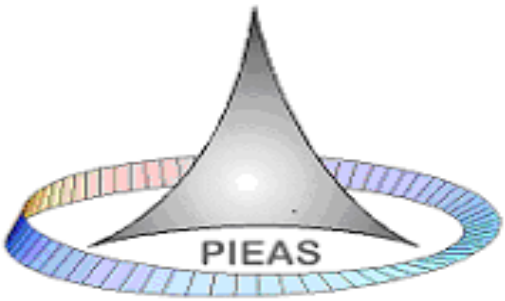
Vital Sign: Personal Signature Based Biometric Authentication System



Muhammad Nauman Sajid

(This thesis is submitted in partial fulfillment of requirements for the BS Degree in Computer Information Sciences)

Pakistan Institute of Engineering & Applied Sciences,
Nilore-45650, Islamabad
September, 2009


بسم الله الرحمن الرحيم

Certificate of Approval

Certified that the work contained in this thesis titled as

Vital Sign: Personal Signature Based Biometric Authentication System

was carried out by ***Mr. Muhammad Nauman Sajid*** under my supervision and that in my opinion, it is fully adequate, in scope and quality, for the degree of the BS (Computer and Information Sciences).

Approved By:

Signature: _____

Supervisor Name: Mr. Fayyaz ul Amir Afsar

September 2009

Dedication

"To My Grand Mother, Parents and Younger Brothers who mean the entire world to me"

Acknowledgement

All thanks and gratitude goes to Allah for all the blessings He has bestowed upon us. He has given us everything we have and often we forget about the bounties we enjoy. I would like to pledge my humble regards firstly to Allah Almighty who conferred the determination and strength to remain focused and cohesive during this project.

All respects for our dear Holy Prophet Muhammad (SAW) who enlightened our minds to recognize our Creator and thyself as the last Prophet of Allah & great benefactor of mankind.

I am gratified to my parents for their support and help made be capable of doing this project.

The work would not be carried out so smoothly without the help and support from a number of people. I am profoundly grateful to my supervisor, Mr. Fayyaz ul Amir Afsar Minhas (DCIS, PIEAS) and Co-supervisor Dr. Muhammmad Arif for their continued support and effort. Especially to my supervisor whose grasp on knowledge, disciplined work and appropriate guidelines have been very useful in my whole project.

At the end I would like to thanks all my batch mates and juniors who helped in creation of my database.

Abstract

Signature is a behavioral biometric: it is not based on the physical properties, such as fingerprint or face, of the individual, but behavioral ones. Signature verification is split into two according to the available data in the input. Offline (static) signature verification takes as input the image of a signature and is useful in automatic verification of signatures found on bank checks and documents. Online (dynamic) signature verification uses signatures that are captured by pressure-sensitive tablets that extract dynamic properties of a signature in addition to its shape.

The purpose of project is to develop an authentication system based on personal signatures. Signature verification is an important research topic in the area of biometric verification. In this project the work is done in such a way that the signatures are captured using pen tablet. The signatures are gathered using local features and 1-D Wavelet transform (WT) is performed on that features. The Discrete Cosine Transform (DCT) is used to reduce the approximation coefficients vector obtained by WT to a feature vector of a given dimension. Then the classification is performed using Linear programming descriptor (LPD), Simple K-Nearest Neighbor (kNN) and Pruned fuzzy k-Nearest Neighbor (Pwfknn). The results obtain in this projects are with improved knn FAR and FRR for random forgeries is 14.40% and 3.2% respectively and FAR for skilled forgeries 12.80%, with lpd FAR and FRR on random forgeries is 1.52% and 23.80% respectively and FAR for skilled forgeries is 14.20%, with pwfknn FAR 3.44% and FRR 13.11% on random and FAR of 15.4%, on skilled forgeries.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Humans usually recognize each other according to their various characteristics for ages. We recognize others by their face when we meet them and by their voice as we speak to them. These characteristics are their identity. To achieve more reliable verification or identification we should use something that really characterizes the given person.

1.1 Biometrics:

The term "biometrics" is derived from the Greek words bio (life) and metric (to measure). Biometrics refers to the automatic identification of a person based on his/her physiological or behavioral characteristics. This method of identification is preferred over traditional methods involving passwords and PIN numbers for its accuracy and case sensitiveness. A biometric system is essentially a pattern recognition system which makes a personal identification by determining the authenticity of a specific physiological or behavioral characteristic possessed by the user. These characteristics are measurable and unique. These characteristics should not be duplicable. An important issue in designing a practical system is to determine how an individual is identified. Depending on the context, a biometric system shown in Figure 1-1 can be either a verification (authentication) system or an identification system.

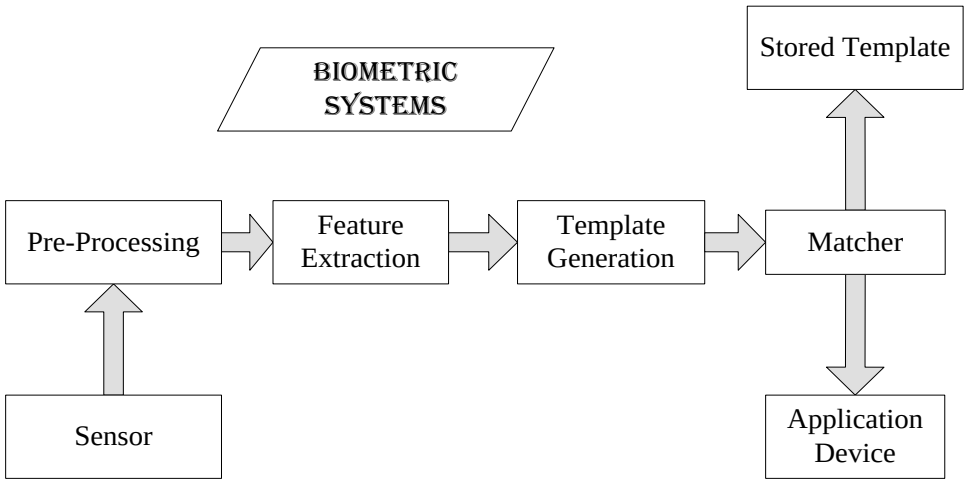


Figure 1-1 Biometrics Authentication System

1.1.1 Identification

This involves establishing a person's *identity* based *only* on biometric measurements. The comparator matches the obtained biometric with the ones stored in the database using a 1:N matching algorithm for identification.

1.1.2 Verification

It involves confirming or denying a person's *claimed identity*. When the user claims to be is already enrolled in the system (presents an ID card or login name). In this case the biometric data obtained from the user is compared to the user's data already stored in the database. The matching algorithm used in this case is 1:1.

1.1.3 Advantages of a biometrics system

The fact that you will have to personally be present in order to authenticate yourself is the advantage of this system. Finger print or retina of the eyes of one person does not match with anyone else's data in the database. Therefore there is absolutely no chance of other people using your identity.

1.1.4 Disadvantages of a biometric system

Biometric system also has some of disadvantages that can be given as:

- The finger prints of those people working in Chemical industries are often affected. Therefore these companies should not use the finger print mode of authentication.
- It is found that with age, the voice of a person differs. Also when the person has flu or throat infection the voice changes or if there there are too much noise in the environment this method may not authenticate correctly. Therefore this method of verification is not workable all the time.
- For people affected with diabetes, the eyes get affected resulting in differences.
- Biometrics is an expensive security solution.

1.2 Signature Verification:

Signature verification is a common behavioral biometric to identify human beings for purposes of establishing their identity. Signatures are particularly useful for identification because each person's signature is highly unique, especially if the dynamic properties of the signature are considered in addition to the static shape of the signature. Even if skilled forgers can accurately reproduce the shape of signatures, it is unlikely that they can simultaneously reproduce the dynamic properties as well.

1.2.1 Types of Signature verification

The purpose is to extract information of handwriting to establish the identity of the signer. Signature verification is split into two according to the available data in the input.

Offline (Static): Offline (static) signature verification takes as input the image of a signature and is useful in automatic verification of signatures found on bank checks and documents. Some samples of offline signature shown in Figure 1-2.

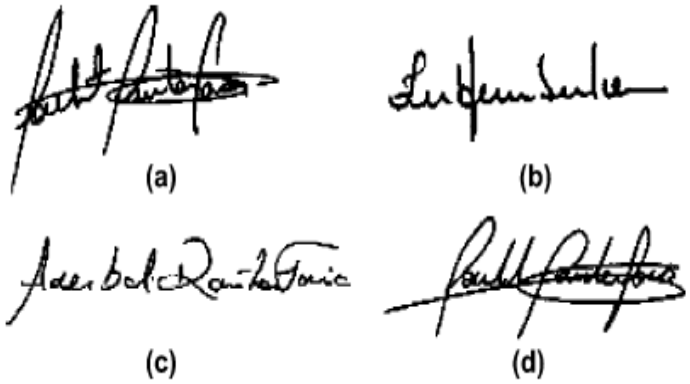


Figure 1-2 Offline Signatures

Online (Dynamic): Online (dynamic) signature verification uses signatures that are captured by pressure-sensitive tablets (shown in Figure 1-3) that extract dynamic properties of a signature in addition to its shape, and can be used in real time applications like credit card transactions, protection of small personal devices (e.g. PDA, laptop), authorization of computer users for accessing sensitive data or programs, and authentication of individuals for access to physical devices or buildings.

Figure 1-3 Online Signatures

1.2.2 Why Online (Dynamic)

Signatures in off-line systems usually may have noise, due to scanning hardware or paper background, and contain less discriminative information since only the image of the signature is the input to the system. While genuine signatures of the same person may slightly vary, the differences between a forgery and a genuine signatures may be imperceptible, which make automatic off-line signature verification be a very challenging pattern recognition problem. Besides, the difference in pen widths and unpredictable change in signature's aspect ratio are other difficulties of the problem. Worth to notice is the fact that even professional forensic examiners perform at about 70% of correct signature classification rate (genuine or forgery).On-line signatures are more unique and difficult to forge than their counterparts are, since in addition to the shape information, dynamic features like speed, pressure, and capture time of each point on the signature trajectory are available to be involved in the classification. In other words, on-line signatures have an extra dimension, which is not available for the off-line signatures. As a result, on-line signature verification is more reliable than the off-line.

Performance Evaluation of Signature vs. System: In evaluating the performance of a signature verification system, there are two important factors: the false rejection rate (FRR) of genuine signatures and the false acceptance rate (FAR) of forgery signatures. As these two are inversely related, lowering one often results in increasing

the other. Usually we talk about the equal error rate (EER) which is the point where FAR equals FRR.

There are two types of forgeries:

- A ***skilled*** forgery is signed by a person who has had access to a genuine signature for practice.
- A ***random*** or ***zero-effort forgery*** is signed without having any information about the signature, or even the name, of the person whose signature is forged.

The performance of the available on-line signature verification algorithms lies between 1% and 10% equal error rate, while off-line verification performance is still between 70% and 80% equal error rate [11].

There have been several studies on on-line signature verification problem. On-line signature verification systems differ on various issues, such as data acquisition, preprocessing, and dissimilarity calculation.

1.2.3 Advantages:

Signature verification presents three likely advantages over other biometrics techniques from the point of view of adaptation in the market place.

- First it is a socially accepted identification method already in use in banks and credit card transaction.
- Second, most of the new generation of portable computers and personal digital assistants (PDAs) use handwriting as the main input channel.
- Third, a signature may b changed by the user; similarly to a password while it is not possible to change finger prints iris or retina patterns.
- The cost to employ that particular biometric data.

Therefore, automatic signature verification has the unique possibility of becoming the method of choice for identification in many types of electronic transactions.

1.2.4 Applications:

Signature verification has been and is used in a number of applications ranging from governmental use to commercial level to forensic applications. A few of them are discussed below:

Security for Commercial Transactions: Nowadays signature verification is swiftly penetrating into commercial use. It can be used for authentication on ATMs, point of sales or for package delivery companies. The internationally recognized courier service UPS has been using signature verification for many years now for personnel authentication.

Secure computer system authentication: Logging on to PCs could be done with a combination of signature verification system and fingerprint identification system to achieve a higher level of security in a more sensitive area. We can also use a combination of password and signature verification system. This would allow the employees to have a higher level of security and confidentiality for their clients and protection of their trade secrets.

Cheque Authentication: Signatures have been used for decades for cheque authentication in banking environment. But even experts on forgeries can make mistakes while authenticating a signature. Off-line signature verification can be used for cheque authentication in commercial environment to enhance security.

Forensic Applications: Signature verification techniques have always been used for cheque fraud and forensic applications over the years.

1.3 General System Overview:

A dynamic signature verification system gets its input from a digitizer or other, usually pen-based, dynamic input device. The signature is then represented as one or several time-varying signals. In other words, the verification system focuses on how the signature is being written rather than how the signature was written. This provides a better means to grasp the individuality of the writer but fails to recognize the writing itself.

1.3.1 General Diagram:

The system design for this project has different phases. These phases are treated as an individual processes. The general system diagram for signature verification is as given below in Figure 1-4:

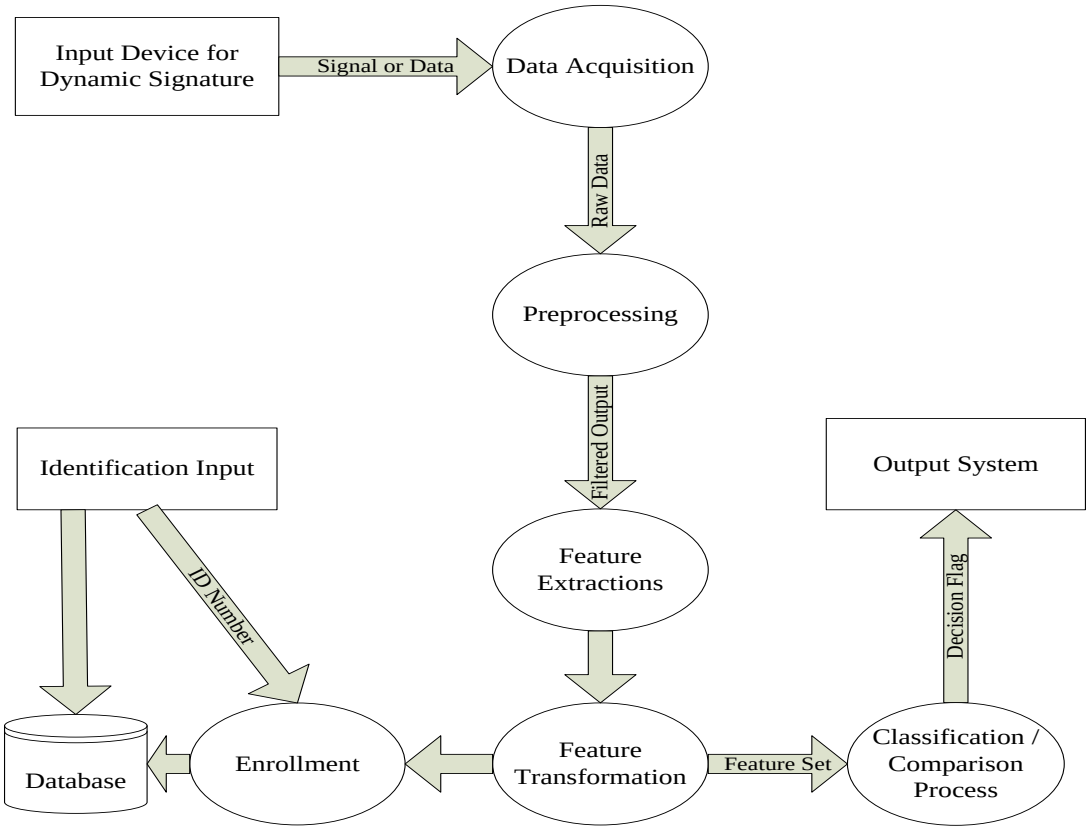


Figure 1-4 General System Overview

1.3.2 Input:

The input to an on-line signature verification system is dynamic. This input is normally taken through a digital tablet. This input is digitized and fed into the computer for processing. First of all pre-processing is done on the input received and then some features are extracted on the basis of which the signature is validated.

1.3.3 Output:

The output required from an online signature verification system is a decision if the person providing the signature is authorized.

1.3.4 Preprocessing:

There are some commonly done preprocessing steps, aimed to improve the verification performance of a system. These range from size normalization to smoothing of the trajectory and re-sampling. Tablets with low resolutions or low sampling rates may give signatures that have jaggedness which is commonly removed using smoothing techniques. In the systems where tablets of different active areas are

used, signature size normalization is a frequently used preprocessing technique. Comparing two signatures having the same shape but different sizes would result in low similarity scores. Size normalization is commonly applied to remove that affect. Modern tablets have a sampling rate of more than 100 trajectory points per second. In some of the previous methods, re-sampling, as a preprocessing step, was used to get rid of possibly redundant data. After successful re-sampling, shape related features were more reliably extracted

1.3.5 Feature Extraction:

Feature extraction phase is one of the crucial phases of an on-line signature verification system. Features may be classified as global or local, where global features identify signature's properties as a whole and local ones correspond to properties specific to a sampling point. As an example, signature bounding box, trajectory length or average signing speed are global features, and distance or curvature change between consecutive points on the signature trajectory are local features.

1.3.6 Enrollment

During enrollment signature are stored against each user. The Non skilled forgeries and skilled forgeries are also stored in the database.

1.3.7 Verification:

During the verification phase, a test signature and an ID of a claimed user are submitted to the system. The test signature is compared with the template of reference signatures generated in the previous process. A threshold value is defined and the signature is classified as genuine or forged depending on the threshold..

1.3.8 Identification:

During the identification phase, only the test signature and no ID are submitted to the system. The unknown test signature is compared with every template signature in the database. The signature is identified belonging to a single class of signatures in the database to which it is closest to.

1.4 Thesis Outline

In chapter 2, a comprehensive literature survey of the major techniques implemented in the field of signature verification is presented. In chapter 3 the proposed system is discussed along with Database creation. Results of the implemented and optimized techniques are also discussed. In chapter 4 the discussion of the results are included. the references and appendix is at the end of the thesis.

Chapter 2. Literature Survey

In human life security is one of the greatest issues. It's the basic fundamental of all systems developed nowadays. Biometric authentication system got a lot of importance. As they are secure, easy to use, easy to develop, uses basic techniques of signal processing and cheap to build. This results the familiarity of biometric authentication system. In these techniques signature verification is the most famous one because of cheap data acquisition devices. One can see the use of on-line signature verification in many real time applications, such as credit card transactions, document flow applications, and identity authentication prior to access of sensitive resources. There have been several studies on on-line signature verification problem. On-line signature verification systems differ on various issues, such as data acquisition, preprocessing, and dissimilarity calculation.

Most commonly used on-line signature acquisition devices are pressure sensitive tablets with or without visual feedback. Smart pens capable of measuring forces at the pen-tip, exerted in three directions, are also widely used in signature verification systems. Special hand gloves with sensors for detecting finger bend and hand position and orientation, and a CCD camera based approaches were also in signature acquisition; however, due to their cost and impracticality, such devices couldn't find place in real systems. Depending on the device used, fair amount of preprocessing may be applied to a signature data prior to the feature extraction phase.

This portion of thesis is about the previous work in the field of signature verification. As we know that signature verification can be categorized in two fields. Online and offline, but we will only discuss the online techniques as our interest of study is online signature verification. The on-line signature verification techniques can be classified into two broad areas.

1. Using features selected from the visible parts of the signature (the parts that are actually drawn by the signer).

2. Using features selected from virtual strokes or invisible parts of the signature (the parts that are not created but are imagined to be created during the pen up time).

2.1 Using Variable Length Segmentation and Hidden Markov Models:

In this research paper Shafiei [6] introduced a new on-line handwritten signature verification system using Hidden Markov Model (HMM) is presented. The system he proposed is based on variable length segmentation of signatures in a HMM model for on-line signature verification. To achieve this he segmented each signature based on its perceptually important points. Then after some preprocessing, he associated to each segment a scale and displacement invariant feature vector. The result of segmentation is a number of segments for each signature. Each segment is characterized by location of its most significant point in the signature, average velocity, average acceleration, average pressure, pressure variance and two angles of tangent lines to curve of segment in two segment end points. The Figure 2-1 shows the angle of tangent lines at two end points.

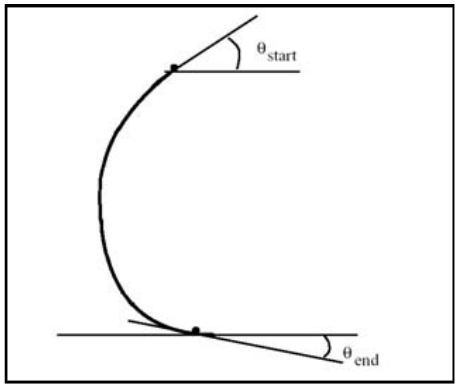


Figure 2-1 Ref [6] Angle of tangent lines at two end points

Finally, the resulted sequence is then used for training an HMM to achieve signature verification. For each signer *i*, an HMM is trained using 5 genuine signatures of *i*. Assuming mixture of ten Gaussians for emission probabilities of this HMM. The number of states of each HMM model equals 0.5 times the average number of segments that in segmentation step is computed for each signature in the training set. He used EM algorithm during training and the Viterbi algorithm during the verification phase to approximate the likelihood of the signature. The overall information of that paper is shown in Table 2-1.

Table 2-1 On Using Variable Length Segmentation and HMM

Features used	Database Size				Features Extracted	Results	
	Total Persons	No. of sig/person	Forgeries	Total Sign		FAR	FRR
➢ Left to right HMM with loop ➢ Forward and skip transitions ➢ Density function modeling	69	4-34	1010	622	➢ Location of most significant point in the signature ➢ Average velocity ➢ Average acceleration ➢ Average pressure ➢ Pressure variance ➢ Two angles of tangent lines to curve	4%	12 %

The high FRR in this case, comparing to other works, were caused by the small number of signatures used in our training phase. In spite of using Gaussian mixtures for modeling interpersonal variability, the HMM doesn't learn adequately these variability when using small number of signatures in the training phase.

2.2 On-line Handwritten Signature Verification using HMM Features:

In this paper Kashi [7] proposed a method for the automatic verification of on-line handwritten signatures using both global and local features. The global and local features capture various aspects of signature shape and dynamics of signature production. He demonstrated that with the addition to the global features of a local feature based on the signature likelihood obtained from Hidden Markov Models (HMM), the performance of signature verification improved significantly. In this paper he models the signing process with several states that constitute a Markov chain, each of them corresponding to a signature segment. The states are not directly observable (hidden); one can only observe the signature local features (such as tangent angles). In this signature modeling, the handwriting tangent and its derivative

as an observation vector in equal length segmentation is used. For this observation vector, the HMM likelihood method of the signature verification performed comparable to the Euclidean distance rule. The detailed information of that research paper is shown in Table 2-2.

Table 2-2 On-line Handwritten Signature Verification using HMM Features

Features used	Database Size				Features Extracted	Results	
	Total Persn	No. of sig/persn	Forgrs	Total Sign		FAR %	FRR %
➢ Length-to-width ratio L ➢ Horizontal span ratio ➢ Horizontal centroid ➢ Vertical centroid	59	6	325	542	Total of 23 Global features ➢ Total signature time ➢ Time down ratio ➢ x , y components of velocity and acceleration ➢ Root-mean-square (rms) speed V ➢ Average horizontal speed V. ➢ Integrated centripetal acceleration	13- 5	1

The combination of the HMM log and global feature information improved the performance of the system when compared to either the local or global methods used independently. The equal error rate decreased from about 4.5% to about 2.5% with the enhanced technique. At the 1% false rejection (FR) point, the addition of the local information reduce the false acceptance.

2.3 Dynamic Signature Verification using Local and Global Features:

In this paper Pippin[5] presented two verification filters, each employing different techniques commonly used in the literature. The first filter extracts high-level global features of a signature and compares these against stored signature templates using KNN classification. The second filter uses velocity based stroke segmentation to encode the signature as a series of strokes and then uses dynamic time warping to find the closest matches between test and template signatures.

Table 2-3 Dynamic Signature Verification using Local and Global Features

Features used	Database Size				Features Extracted	Results	
	Total Persns	No. of sig/person	Forgrs	Total Sign		1st Filter	2nd Filter
➢ Average Pressure ➢ Pen Tilt. ➢ Average Velocity ➢ Number of Pen Ups ➢ Number of Strokes ➢ Velocity as a function of time	19	10	73	180	➢ Average Pressure ➢ Pen Tilt ➢ Average Velocity ➢ Number of Pen Ups ➢ Number of Strokes	91%	77%

Considering only global features of a signature has advantages that it is simple to compute and addresses privacy concerns because it does not need to retain the original signature once the features have been extracted. This made it ideal as an inexpensive technique that can be used to catch a majority of forgeries, without risk to privacy. With a small number of global features, this technique can classify signatures with approximately 89% accuracy. Strength of this approach is that as an individual's signature changes over time, each signature need only be added to the reference database, and newer signatures will naturally be closer to more recent reference signatures. The detailed information of that research paper is shown as follow:

Two techniques, using dynamic global and local features, for online signature verification were described. It was also shown that signer specific thresholds improved the performance of the local filter. Moving forward, further experimentation on a larger dataset should be performed. However, it is expected that with additional experimentation and adjustment of the feature sets, improved results can be obtained.

2.4 New extreme points warping technique:

In this paper, Feng[8] proposed a new warping technique for the functional approach in signature verification. The commonly used warping technique is dynamic time warping (DTW). As we know that there are two common methodologies to verify signatures: the functional approach and the parametric approach So the functional based approach was originally used in speech recognition and has been applied in the field of signature verification with some success since two decades ago. The new warping technique he proposed is named as extreme points warping (EPW). It proved to be more adaptive in the field of signature verification than DTW, given the presence of the forgeries. In the functional approach, a straightforward way to compare two signal functions is to use a linear correlation but a direct computation of the correlation coefficient is not valid due to the following two problems:

- Difference of overall signal duration.
- Existence of non-linear distortions within signals.

For a signal function, it is unlikely that the signal duration is the same for different samples even from the same signer. In addition, for different signings, distortions occur non-linearly within the signals. To correct the distortion, a non-linear warping process needs to be performed before comparison. An established warping technique used in speech recognition is dynamic time warping, or DTW. For the past two decades, the use of DTW has also become a major technique in signature verification. Though DTW has been applied to the field with some success, it has some drawbacks. DTW has two main drawbacks when applied in signature verification:

- Heavy computational load,
- Warping of forgeries.

The first drawback is a known problem in speech recognition. This is because DTW performs nonlinear warping on the whole signal. The execution time is proportional to the square of the signal size. To reduce the computation time, define boundary conditions in the DTW matching matrix. The second drawback, however, is not well documented in the past literature, but still got good accuracy and results as mentioned below in Table 2-4:

Table 2-4 New extreme points warping technique

Features used	Database Size				Features Extracted	Results	
	Total Persns	No. of sig/persn	Forgrs	Total Sign		EER (EPW)	EER (DTW)
➢ Rise distance w.r.t time ➢ Drop distance w.r.t time	25	30	250	1000	Variations ➢ Non-synchronicity for the start point ➢ Existence of ripples ➢ Non-synchronicity for the end point	27.7 %	35%

A new warping technique call EPW replaced the commonly used DTW. Instead of warping the whole signal as DTW does, EPW warps a set of selective points, i.e. the EPs on the signal. Through matching the EPs and warping the segments linearly, we achieve the goal of warping the whole signal. Since EPW warps only EPs, the local curvatures between the EPs are preserved, which prevents forged signals taking advantages from the warping process. With the adoption of EPW, the EER is improved by a factor of 1.3 over using DTW and the computation time is reduced by a factor of 11. Hence the new technique, EPW, is quite promising to replace DTW to warp signals in the functional approach, as part of a more effective signature verification system.

2.5 Wavelet Transform Based Global Features:

In this paper a system proposed by F.A. Afsar[10], U. Farukh and M Arif which worked in such a way that first the global features are extracted from the spatial coordinates and these features are by obtained during the data acquisition phase. The method used is one dimensional wavelet transform. Then using K-NN classifier the results are obtained and proved the accuracy of the proposed technique. It's a global feature based approach to signature verification. The signature patterns are matched based on wavelet domain features that are extracted from the normalized spatial coordinates of the signatures obtained during signature acquisition. The differences between the spatial coordinates of consecutive points in the signature are also

subjected to wavelet decomposition and feature extraction. The total temporal duration of the signature as a distinguishing feature during classification is also used. The Figure 2-2 shows the block diagram of the system.

The system is described in these phases

- Acquisition
- Preprocessing
- Feature Extraction
- Template Generation
- Feature Matching

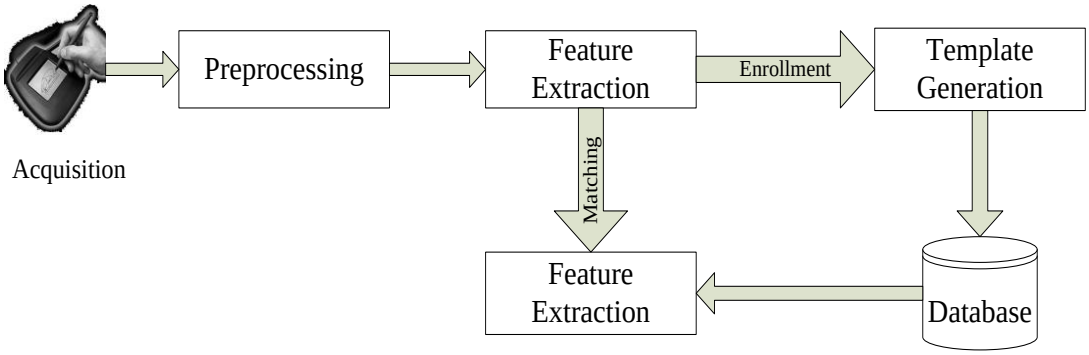


Figure 2-2 Ref Afsar [10]: System Overview

Online signatures are generally acquired by using pressure sensitive tablets. Preprocessing is carried out prior to feature extraction in order to improve the reliability and accuracy of the feature extraction process. Then the features are extracted using the local and global properties. During enrollment phase of an online signature verification system, features from multiple training signatures of a subject are used to create a template for the subject. The template is stored in a database and is used later in the matching phase. In the matching phase of an online signature verification system, features extracted from a given signature are compared with the stored template to generate the matching score on which the verification decision is based.

These results very clearly demonstrate the effectiveness of the global features obtained using the Wavelet Transform. The results can be further improved if orientation normalization and re-sampling is carried out during preprocessing and some local features are also used along with the global ones. The detailed information is shown in Table 2-5.

Table 2-5 Wavelet Transform Based Global Features:

Features used	Database Size				Features Extracted	Results	
	Total Persns	No. of sig/persn	Forgrs	Total Sign		FAR	FRR
➢ Pressure ➢ Velocity ➢ Pen Ups ➢ Velocity as a function of time ➢ X-coord ➢ Y-coord	100	15	5	2000	➢ Total time ➢ No of zero crossings in x-velocity ➢ No. of zero crossings in y-velocity ➢ No. of zero crossings in x-acceleration ➢ No. of zero crossings in y-acceleration ➢ No. of zero values in x-acceleration ➢ No. of zero values in y-acceleration ➢ Average pressure ➢ overall path length	Ran 3.21 Skl 6.79	Ran 3.27 Skl 6.61

2.6 Two-Stage Statistical Model:

In this paper, Liang Wan [15] proposed a new two-stage statistical system for automatic on-line signature verification. System is composed of a simplified GMM model for global signature features, and a discrete HMM model for local signature features. He introduced specific simplification strategies for model building and training. System requires only 5 genuine samples for new users and relies on only 3 global parameters for quick and efficient system tuning. Experiments are conducted to verify the effectiveness of our system.

It is basically a two-stage statistical system for on-line signature verification. System is composed of a simplified GMM model built on global signature properties and a left-to-right HMM model based on segmental features. The general GMM model and HMM model are complex for this specific application, so he introduced specific strategies to do model simplification and initialization. System depends on three global parameters to control its performance. Parameters are estimated globally for all users such that forgeries are only needed for system tuning.

For each signer, two models are processed separately, corresponding to global and local signature information. In global modeling, a Gaussian mixture to estimate

the distribution of global features is used, such as time duration and average speed. In local modeling, he built an HMM model based on both piecewise information and structural relation between strokes. The Figure 2-3 shows the block diagram of signatures verification system used in the paper.

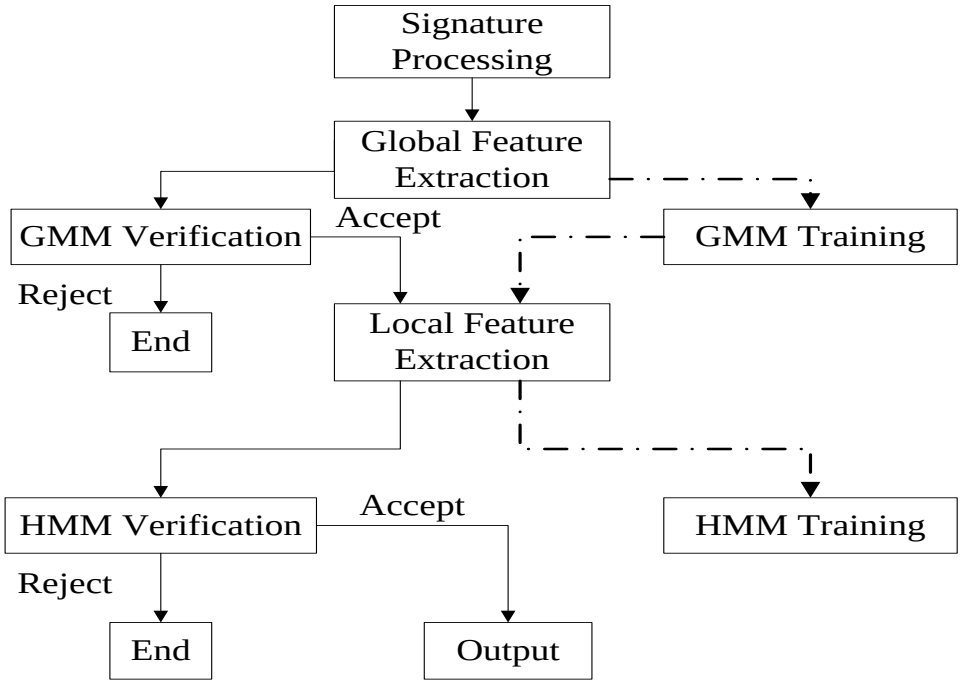


Figure 2-3 Ref [15] Block diagram of signature verification system

In the Figure 2-3 the left part (indicated by solid lines) shows the verification procedure. The input signature is first fed into the GMM classifier. If the confidence is below a threshold, the signature is rejected as a forgery. For signatures that pass the first test, we extract segmental feature sequences and feed them into the HMM classifier.

The signature is accepted as genuine when it also passes the HMM verification test. Listing the highlights of system in the following:

- Given the well-established system, it only uses few genuine signatures as training data for a new user. No forgeries are needed in the training stage.
- Discriminative features are proposed at global and local levels, respectively.
- Our system adopts a two-stage statistical structure, where the global level features can rule out obvious forgeries quickly.

The system can be easily tuned since there are only three global parameters involved.

Table 2-6 Two-Stage Statistical Models

Features used	Database Size				Features Extracted	Results
	Total Persns	No. of sig/persn	Forgrs	Total Sign		Accuracy
➢ the average speed ➢ maximum speed ➢ average pressure ➢ maximum pressure difference between two sample points, ➢ total duration time ➢ Ratio of pen-down time to total writing time.	NA	5	No	5/person	➢ width and height ➢ total length of signature strokes ➢ stroke count and number of self-intersection points; ➢ segment count ➢ Total curvature.	93.3 % (With Pressure) 89.7% (Without Pressure)

2.7 Biometric Authentication using Online Signatures:

In his paper Alisher [16], presented a system for on-line signature verification, approaching the problem as a two-class pattern recognition problem. During enrollment, reference signatures are collected from each registered user and cross aligned to extract statistics about that user's signature. A test signature's authenticity is established by first aligning it with each reference signature for the claimed user. The signature is then classified as genuine or forgery, according to the alignment scores which are normalized by reference statistics, using standard pattern classification techniques. He experimented with the Bayes classifier on the original data, as well as a linear classifier used in conjunction with Principal Component Analysis (PCA). The system has following phases:

- Data Acquisition
- Feature Extraction
- Signature Alignment
- Enrollment
- Training

- Verification

The Table 2-7 gives the detailed information about the paper

Table 2-7 Biometric Authentication using Online Signatures

Features used	Database Size			Features Extracted	Results	
	Genuine Sign	Forgrs	Total Sign		FAR	FRR
➢ X-coordinates ➢ Y-coordinates	182	313	500	➢ x-y coordinates relative to the first point of signature trajectory ➢ x and y coordinate differences between two consecutive points(¢*x;¢y*), ➢ Curvature differences between consecutive points.	Skl **Bayes** 3.51% **PCA** 1.28 %	Gen **Bayes** 2.19 % **PCA** 1.65%

During the enrollment phase, the user supplies a set of reference signatures which are used to determine user dependent parameters characterizing the variance within the reference signatures. The reference set of signatures, together with these parameters, are stored with a unique user identifier in the system's database. When a test signature is input to the system for verification, it is compared to each of the reference signatures of the claimed person. The person is authenticated if the resulting dissimilarity measure is low, rejected otherwise.

2.8 Signature Recognition through Spectral Analysis:

In this research by CMAN F. LAM [17], the signatures were normalized for size, orientation, etc. After normalization, the X and *Y* coordinates of each sampled point of a signature over time (to capture the dynamics of signature writing) were represented as a complex number and the set of complex numbers transformed into the frequency domain via the fast Fourier transform. A Gaussian probabilistic model was developed to screen and select from the large set of features (e.g. amplitude of each harmonics). The significant harmonics of the signature were sorted according to the chi-square value, which is equivalent to the signal-to-noise ratio. Fifteen harmonics with the largest signal-to-noise ratios from the true signatures were used in a discriminant analysis. The Table 2-8 gives the detailed information about the paper.

Table 2-8 Signature Recognition through Spectral Analysis

Features used	Database Size				Features Extracted	Results
	Total Persons	No. of sig/person	Forgeries	Total Signatures		Error
➢ shape, ➢ motion ➢ pressure ➢ timing, ➢ transformation methods	20	8	152	312	➢ Shape ➢ Motion ➢ Pressure ➢ Timing, ➢ Transformation methods	2.5%

Signature data were recorded as integer values on a digital graphic tablet at intervals of 10 ms for 1024 points**.** The values of X and *Y* ranges from 0 to 2047. The Z values indicate whether the pen is down (Z = 1) or up (Z = 0). The data are stored on the computer in files of length 1024 lines. The recorded signature needs to be preprocessed to remove noise, and minor elements. Which include Spike and Minor Element Removal, Ligature, Drift, position, Duration, rotation, connect tails and scaling. After the signature data were normalized, as discussed in the previous sections, the data were then transformed into the frequency domain via the fast Fourier transform**.**

Chapter 3. Implemented Technique

The proposed technique contains an on-line signature verification system based on local information and on one-class classification using the Linear Programming Descriptor classifier (LPD), k Nearest Neighbor (knn) and Pruned fuzzy k Nearest Neighbor (pwfknn). The information is extracted as time functions of various dynamic properties of the signatures, then the discrete 1-D wavelet transform (WT) is performed on these features. The Discrete Cosine Transform (DCT) is used to reduce the approximation coefficients vector obtained by WT to a feature vector of a given dimension. The classifiers are trained using the DCT coefficients.

3.1 Database Creation

A database was created taking signatures from the students and employees of PIEAS. Signatures were collected from a total of hundred persons with fifteen signatures from each person. The tablet used was WACOM Graphire 4 with a sampling rate of 100 samples per seconds. So a total of 1500 signatures were collected to create the original signature database. A single signature was stored in a single text file.

Five skilled forgeries of each subject were performed and stored to form the forgeries database. Skilled forgeries are the forgeries that are performed by first training the forger to copy the exact dynamics of the original signer.

The forgeries are created by capturing the dynamics of signature. We took the signer in confidence. The simple forgeries are the forgeries which are captured by just seeing the signature. These forgeries are used to train the datasets. As the skilled forgery is a very complex task, so to perform forgeries dynamics of every signature is observed and forgeries are performed.

Some of the original Signatures (Figure 3-1 and Figure 3-3) and their forgeries (Figure 3-2 and Figure 3-4) are shown below.

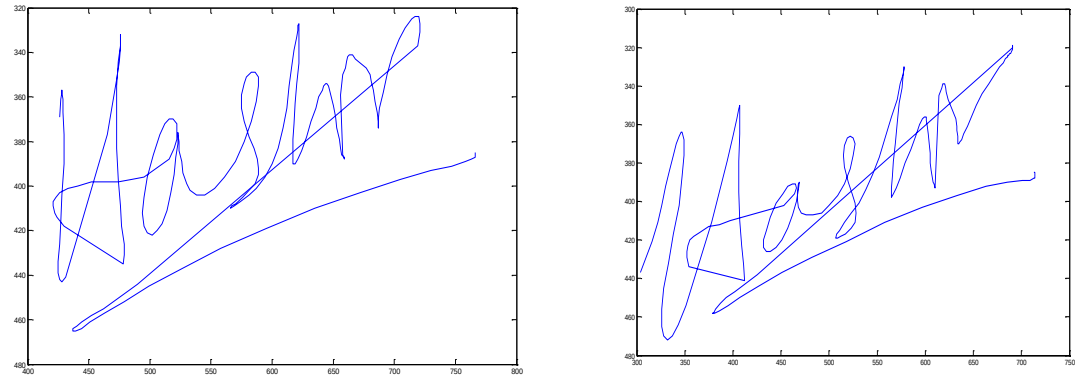


Figure 3-1 Subject 84 Genuine Signatures

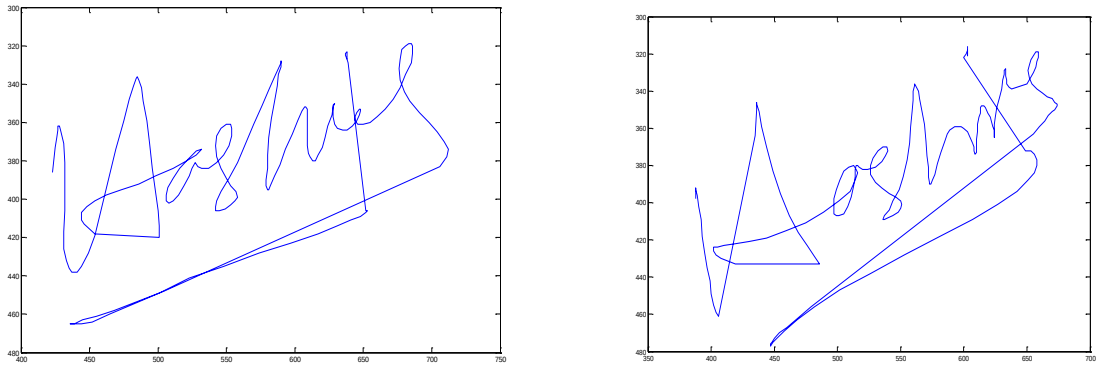


Figure 3-2 Subject 84 Skilled Forgeries

These Figures illustrate the differences between the genuine and fogered signatures. The Figures shows that the criteria for creation of forgeries so strict that there is very little differnce between genuine and forgered signatures.

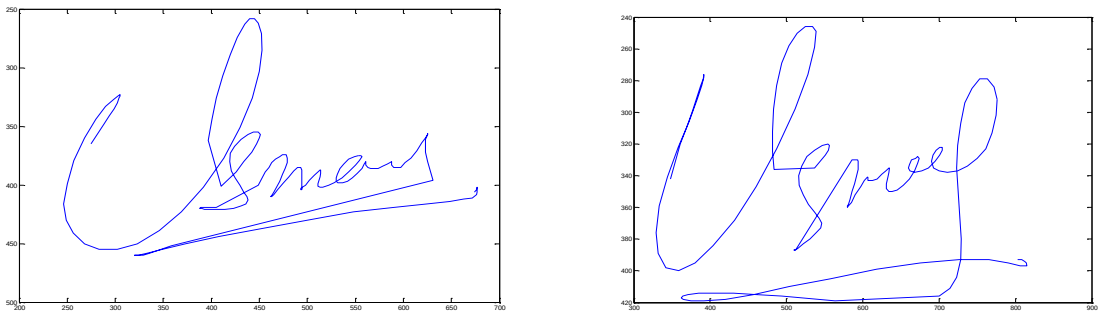


Figure 3-3 Subject 99 Genuine Signatures

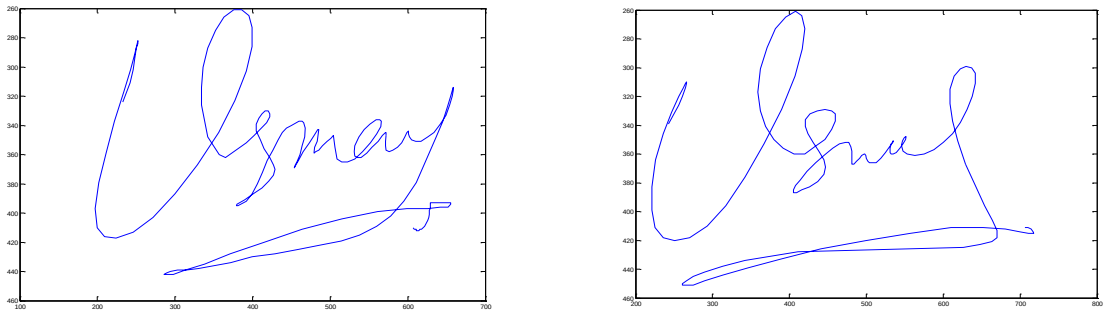


Figure 3-4 Subject 99 Skilled Forgeries

So the database was completed with a comprehensive set of signatures of 100 subjects including 15 original and 5 forged signatures for each person, making a total of 2000 signatures.

3.2 System Overview:

On the basis of research work there are number of system one can suggest fo signature verification in our case the the system proposed has five phases. The system overview is shown in Figure 3-5.

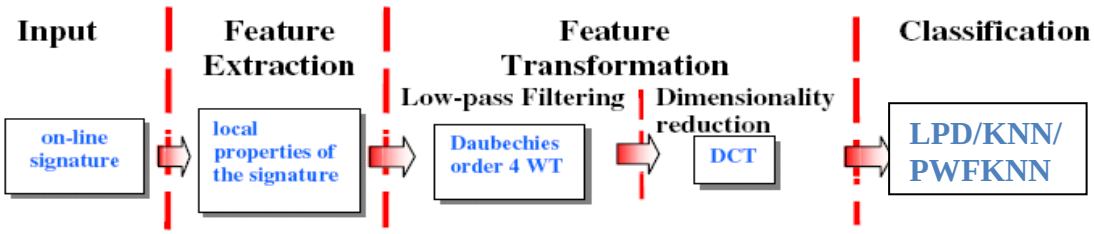


Figure 3-5 Ref [9] System Overview

3.3 Feature Extraction

The input to an on-line signature verification system is dynamic. This input is normally taken through a digital tablet. The acquisition device used is a WACOM Graphire4 pen tablet. The specification of tablet is:

Tablet Dimensions: 8.20" x 8.03" x .70"

Active Area: 3.65" x 5.02"

Pressure Levels: 512

Sampling Rate: 100 trajectory points per second

Feature extraction phase is one of the crucial phases of an on-line signature verification system. Features may be classified as global or local, where global features identify signature's properties as a whole and local ones correspond to properties specific to a sampling point. As an example, signature bounding box, trajectory length or average signing speed are global features, and distance or curvature change between consecutive points on the signature trajectory are local features. Features may also be classified as spatial (related to the shape) or temporal (related to the dynamics). There is no preprocessing done with signatures. Preprocessing is done if have limited sampling rate, limited area for signatures, limited time to capture signatures and similar things like that. The features of each signature are extracted on the basis of local properties. The following 6 local properties are used in features extraction:

1. Horizontal x position trajectory.
2. Vertical y position trajectory.
3. Pressure sequence.
4. Y coordinates differences between two consecutive points (these features correspond to the change in the y direction).
5. X coordinates differences between two consecutive points (these features correspond to the change in the x direction).
6. Pressure differences between two consecutive points (these features correspond to the change in the Pressure value).

Later as mentioned in F.A. Afsar [10], U. Farukh and M Arif research paper the combination of more local properties are used. These are 9 in numbers which are mentioned as follow:

1. Total time, i.e., the number of samples in the list (from first pen-down to last pen-up) with the sampling rate as common time-base.
2. Number of zero crossings in x-velocity, i.e., the number of sign changes in the differences in the pair over the x coordinates.
3. Number of zero crossings in y-velocity, i.e., the number of sign changes in the differences of the pair over the y coordinates.
4. Number of zero crossings in x-acceleration, i.e., the number of sign changes in the differences of the pair over the x velocities.

5. Number of zero crossings in y-acceleration, i.e., the number of sign changes in the differences of the pair over the y velocities.
6. Number of zero values in x-acceleration, i.e., the number of samples with a zero x-acceleration value.
7. Number of zero values in y-acceleration, i.e., the number of samples with a zero y acceleration value.
8. The overall pen-up time, i.e., the number of samples with the pen up.
9. The overall path length, i.e., the sum of the Euclidean distances between the samples.

3.4 Feature Transformation:

Each local property extracted is processed by the Daubechies wavelet for low-pass filtering the signal and by the Discrete Cosine Transform (DCT) for reducing its dimensionality. The Daubechies filters maximize the smoothness of the father wavelet (or ‘‘scaling function’’) by maximizing the rate of decay of its Fourier transform.

3.4.1 Wavelet Transform:

A transform which localizes a function both in space and scaling and has some desirable properties compared to the Fourier transform. The transform is based on a wavelet matrix, which can be computed more quickly than the analogous Fourier matrix. In this project Discrete wavelet Transform (DWT) is used for features extraction. The dwt is used as low pass filter to get the information about features with low frequency components. The Daubechies 4 wavelet is used in our case.

The Daubechies wavelet transform is named after its inventor, the mathematician Ingrid Daubechies. The Daubechies wavelets are a family of orthogonal wavelets defining a discrete wavelet transform and characterized by a maximal number of vanishing moments for some given support. With each wavelet type of this class, there is a scaling function (also called father wavelet) which generates an orthogonal multiresolution analysis.

The Daubechies D4 transform has four wavelet and scaling function coefficients. The scaling function coefficients are

$$h_0 = \frac{(1+\sqrt{3})}{(4\sqrt{2})}, h_1 = \frac{(3+\sqrt{3})}{(4\sqrt{2})}, h_2 = \frac{(3-\sqrt{3})}{(4\sqrt{2})}, h_3 = \frac{(1-\sqrt{3})}{(4\sqrt{2})} \quad \text{Eq. 1}$$

Daubechies D4 scaling functions:

$$A_i = h_0 s_{2i} + h_0 s_{2i+1} + h_0 s_{2i+2} + h_0 s_{2i+3} \quad \text{Eq. 2}$$

$$A[i] = h_0 s[2i] + h_1 s[2i+1] + h_2 s[2i+2] + h_3 s[2i+3] \quad \text{Eq. 3}$$

Daubechies D4 wavelet function:

$$A_i = g_0 s_{2i} + g_1 s_{2i+1} + g_2 s_{2i+2} + g_3 s_{2i+3} \quad \text{Eq. 4}$$

$$C[i] = g_0 s[2i] + g_1 s[2i+1] + g_2 s[2i+2] + g_3 s[2i+3] \quad \text{Eq. 5}$$

The Daubechies filters maximize the smoothness of the father wavelet (or ‘‘scaling function’’) by maximizing the rate of decay of its Fourier transform. They are indexed by their length which may be one of 4, 6, 8, 10, 12, 14, 16, 18 or 20. The names of the Daubechies family wavelets are written dbN, where N is the order, and db the "surname" of the wavelet. The db1 wavelet, as mentioned above, is the same as **Haar** wavelet. The Daubechies order 4 wavelet has been used as low-pass filter. The approximation coefficients of the 1th level of decomposition are extracted in order to de-noise the signal. Some of the Daubechies wavelets are shown in the Figure 3-6:

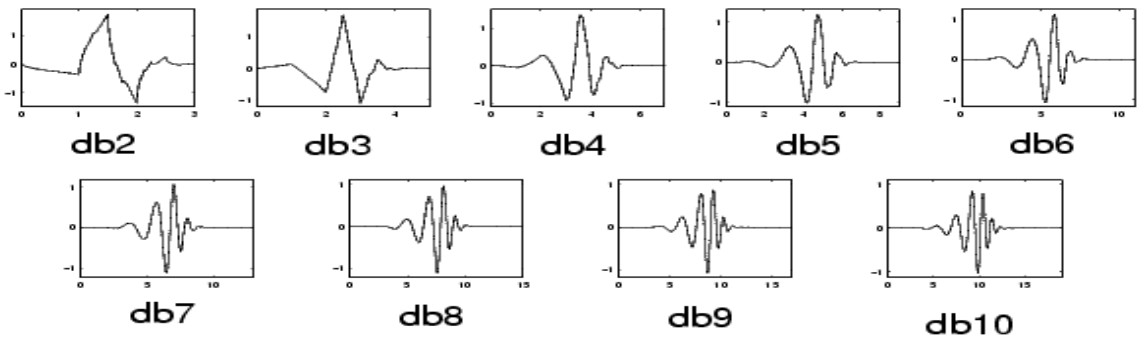


Figure 3-6 N-order Daubechies Wavelet

The Daubechies will affect the signal in a way shown in the Figure. The outputs of the wavelet transform are Approx. Coefficients and Detail Coefficients which are result of Low pass Filtering and High Pass Filtering respectively.

Daubechies order 4 wavelet used as low pass filter. The Figure 3-7-1 shows the mechanism for 1-D wavelet transform.

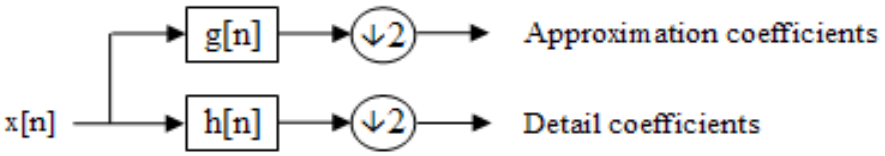


Figure 3-7 1-D Wavelet transform

The effects of the Daubechies wavelet on the signatures are as shown in the Figure 3-8 and Figure 3-9. The Figure 3-8 shows the original signal which is actually local property X coordinate.

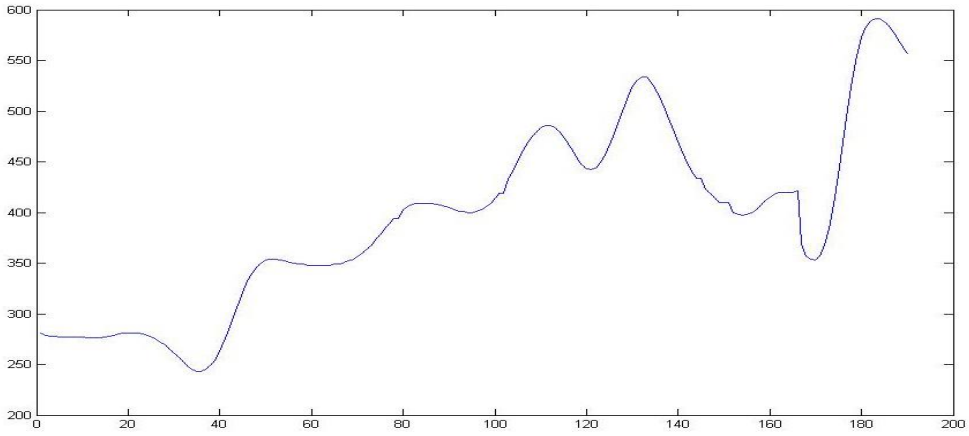


Figure 3-8 X Coordinates of Sig 1-1

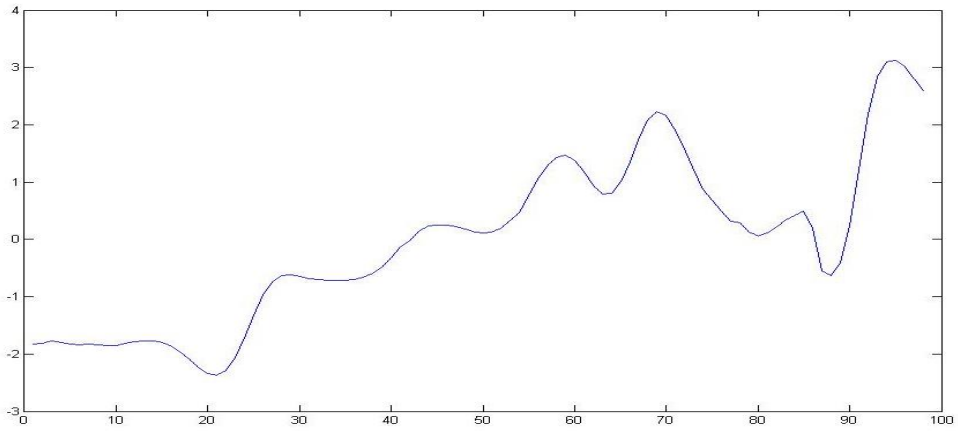


Figure 3-9 Daubechies wavelet of X coordinates

From the above Figures one can easily find that signal became smoother with the wavelet transformation and the jaggedness is removed from the signal. One can

also observe that the y-axis after applying the wavelet also bounded. In this case it's from -3 to +3.

3.4.2 Discrete Cosine Transform (DCT):

In order to reduce the dimensionality of the feature vectors, the Discrete Cosine Transform is directly applied to the approximation coefficients. DCT works better than other well known techniques for dimensionality reduction in this application due to the lack of a large training set. The DCT transform provides a good compromise between information packing ability and computational complexity. Another advantage of the DCT is that most DCT components are typically very small in magnitude because most of the salient information exists in the coefficients with low frequencies.

$$DCT_k(x) = \sum_{n=0}^{N-1} x(n) \cos\left[w_k \left(n + \frac{1}{2}\right)\right] \quad \text{Eq. 6}$$

$$DCT_k(x) = \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi k}{2N}(2n+1)\right], k = 0,1, \ldots, N-1 \quad \text{Eq. 7}$$

The effects of DCT on the signal can be seen in that way. The Figures 3.6 and 3.7 provided the information about x coordinate which a local feature used and the effects of wavelet on that property. Now the effects of DCT on that signal can be seen in Figure 3.10.

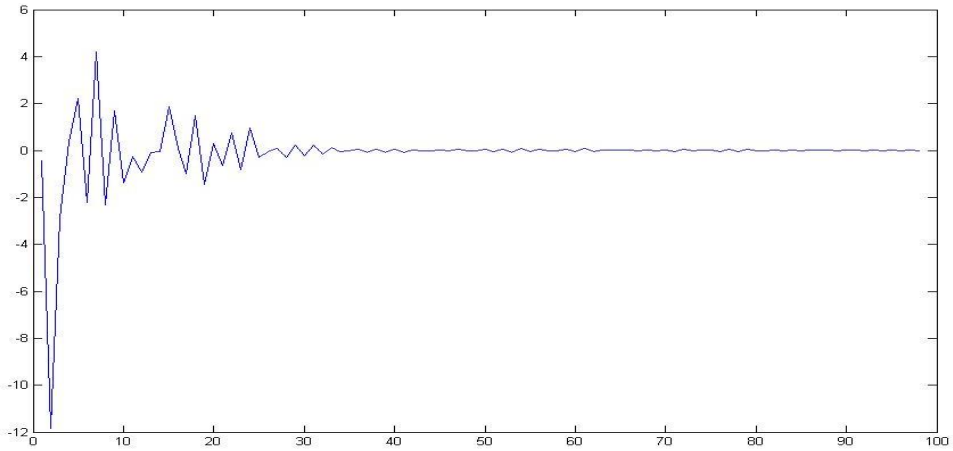


Figure 3-10 DCT of X coordinate

The above diagram showed that DCT transform provides a good compromise between information packing ability and computational complexity. Another advantage of the DCT is that most DCT components are typically very small in magnitude because most of the salient information exists in the coefficients with low frequencies. Truncating, or removing these small coefficients from the representation thereby introduces only small error in the reconstructed images. The coefficients with low frequencies bring the most useful information while the others often bring only noise. The No. of coefficients used in DCT is represented as D which may vary from 10, 13, 15, 17…..

3.5 Classification

Signature verification is a one-class classification problem. Our aim is to verify if the signer is the person that he/she claims to be. The problem in one-class classification is to make a description of a target set of objects. In the signature on-line verification problem an object is a signature, while the individuals are the classes. The difference with conventional classification is that in one-class classification only examples of one class are available. The signatures from a given user are called the target signatures. All the other signatures are per definition outliers. Using a one class classifier we can build a classifier for each individual without any knowledge of the other individuals. For the classification of the signatures we used k-Nearest Neighbor.

3.5.1 K-Nearest Neighbor:

In pattern recognition, the *k*-nearest neighbor algorithm (*k*-NN) is a method for classifying objects based on closest training examples in the feature space. The *k*-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k* nearest neighbors. *k* is a positive integer, typically small. If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. In binary (two class) classification problems, it is helpful to choose *k* to be an odd number as this avoids tied votes.

The neighbors are taken from a set of objects for which the correct classification is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. In order to identify neighbors, the objects

are represented by position vectors in a multidimensional feature space. It is usual to use the Euclidean distance, though other distance measures, such as the Manhattan distance could in principle be used instead. The *k*-nearest neighbor algorithm is sensitive to the local structure of the data. The kNN can be shown best in Figure 3-11.

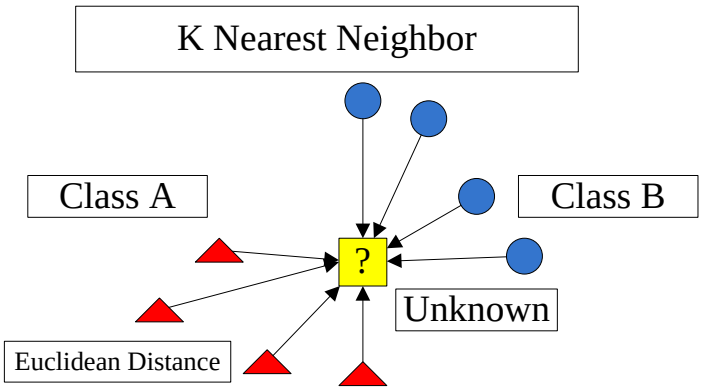


Figure 3-11 K-Nearest Neighbor (k = 3)

Mathematical Model: The k-nearest-neighbor classifier is commonly based on the Euclidean distance between a test sample and the specified training samples. Let $\mathbf{x_i}$ be an input sample with **P** features.

$$x_i = (x_{i1}, x_{i2}, \ldots\ldots\ldots x_{ip}) \quad \text{Eq. 8}$$

n be the total number of input samples

$$(i = 1,2,\ldots\ldots, n) \quad \text{Eq. 9}$$

P the total number of features

$$(j = 1,2,\ldots\ldots, p) \quad \text{Eq. 10}$$

$$x_l = (x_{l1}, x_{l2}, \ldots\ldots\ldots x_{lp}) \quad \text{Eq. 11}$$

$$(l = 1,2,\ldots\ldots, n) \quad \text{Eq. 12}$$

The Euclidean distance between sample $\mathbf{x_i}$ and $\mathbf{x_l}$ where is defined as:

$$d(x_i, x_l) = \sqrt{(x_{i1} - x_{l1})^2 + (x_{i2} - x_{l2})^2 + \ldots\ldots + (x_{ip} - x_{ip})^2} \quad \text{Eq. 13}$$

The k-nearest-neighbor classification rule is to assign to a test point the majority category label of its k nearest training points. In practice, k is usually chosen to be odd, so as to avoid ties. The k = 1 rule is generally called the nearest-neighbor classification rule.

Classification Decisions: Classification typically involves partitioning samples into training and testing categories. Let $\mathbf{x_i}$ be a training sample and **x** be a test sample, and let **w** be the true class of a training sample and $\underline{\mathbf{w}}$ be the predicted class for a test sample, where **k** is the total number of classes. The equation show that all the labels assigned to the target class will belong to a set given below.

$$(w, \underline{w} = 1,2,\ldots..,k) \qquad \text{Eq. 14}$$

During the training process, we use only the true class **w** of each training sample to train the classifier, while during testing we predict the class $\underline{\mathbf{w}}$ of each test sample. It shows that kNN is a "supervised" classification method as it uses the class labels of the training data. *Unsupervised* classification methods, or "clustering" methods, on the other hand, do not employ the class labels of the training data.

The data for classification is divided into two sets training and testing. Training data includes 5 signatures of each signers which makes it 500, while testing data includes the rest of 10 signatures of each signer which makes total testing data equal to 1000. The data is passed to the k-nearest neighbor; the classification resulted in following diagram.

3.5.2 Linear Programming Description (LPD):

Signature verification is a one-class classification problem. The problem of one-class classification is a special type of classification problem. LPD is one-class classifier which is defined later; first we will discuss one-class classifier.

One-Class Classifier: In one-class classification we are always dealing with a two-class classification problem, where each of the two classes has a special meaning. The two classes are called the target and the outlier class respectively:

Target class*:* target class can be defined in a sense that is sampled well, this class has many (training) example objects are available. The sampling of the training set not

necessarily be done completely according to the target distribution, but it might be the user sampled the target class according to his/her idea. It is assumed that the training data reflect the area that the target data covers in the feature space.

Outlier class*:* outlier can be defined in such a manner that the classes or objects that could not be classified as target set, either they were errors or they were very expensive to measure. This class can be sampled very sparsely, or can be totally absent. They may also be very hard to measure. In principle, a one-class classifier should be able to work, solely on the basis of target examples. Another extreme case is also possible, when the outliers are so abundant that a good sampling of the outliers is not possible. The Figure 3-12 shows the OCC.

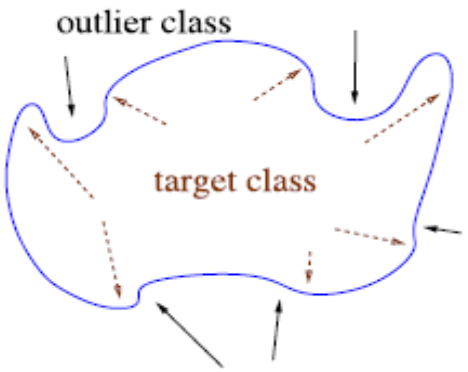


Figure 3-12 One class classification

The one-class classification problem differs in one essential aspect from the conventional classification problem. In one-class classification it is assumed that only information of one of the classes, the target class, is available. This means that just example objects of the target class can be used and that no information about the other class of outlier objects is present. The boundary between the two classes has to be estimated from data of only the normal, genuine class. The task is to define a boundary around the target class, such that it accepts as much of the target objects as possible, while it minimizes the chance of accepting outlier objects.

Our aim is to verify if the signer is the person that he/she claims to be. The problem in one-class classification is to make a description of a target set of objects. In the signature on-line verification problem an object is a signature, while the individuals are the classes. The difference with conventional classification is that in one-class classification only examples of one class are available. The signatures from

a given user are called the target signatures. All the other signatures are per definition outliers. Using a one class classifier we can build a classifier for each individual without any knowledge of the other individuals.

LPD Data descriptor is specifically constructed to describe target objects which are represented in terms of distances to other objects. In some cases it might be much easier to define distances between objects than informative features. The classifier has the following form:

$$f(x) = \sum_i w_i d(x, x_i) \qquad \text{Eq. 15}$$

Where $d(x, x_i)$ is weighted Euclidean distance between x and x_i. Where x and x_i refers to the signatures to be compared. The set $(x_1, \ldots, x_n)$ is the training set. The weights w are optimized such that just a few weights stay non-zero, and the boundary is as tight as possible around the data. LPD depends upon the dissimilarity measure and the proximity mapping.

Dissimilarity representations*:* The basic assumption that an instance belongs to a class is that it is similar to examples within this class. For a dissimilarity measure D, this means that D(r, s) is small if objects r and s are similar, and large if they are different. If we demand that D(r, s) =0, if and only if the objects r and s are identical, this implies that they belong to the same class. This can be extended by assuming that all objects s such that D(r, s) <ϱ for a sufficient small ϱ are so similar to r that they are members of the same class consequently, D(r, t) ≈D(s, t). Objects with large distances are assumed to be dissimilar. The dissimilarity space can be viewed in Figure 3-13.

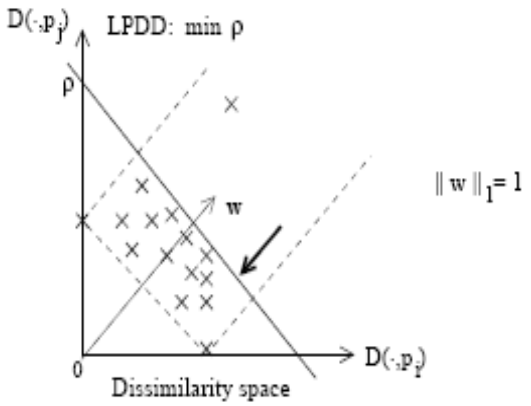


Figure 3-13 LPDD in Dissimilarity Space

When the set R contains objects from the class of interest, then objects z with large D (z, R) are outliers and should be remote from the origin in this dissimilarity space. This characteristic will be used in our One Class Classifier (OCC).

A class represented by dissimilarities can be characterized by a linear proximity function with the weights wj and the threshold ρ. Our one-class classifier CLPDD, (Linear Programming Dissimilarity-data Description) is then defined as:

$$C_{LPDD}(D(z,.)) = I(\sum_{w_j \neq 0} w_j D(z, p_i) \leq p) \qquad \text{Eq. 16}$$

where I is the indicator function. The proximity function is found as the solution to a soft margin formulation. The classification of the signatures is done using LPD. The classifier reported the following results against DCT 10.

3.5.3 Pruned Fuzzy k-Nearest Neighbor Classifier (Pfknn):

This technique is based on supervised learning using k- nearest neighbor. A sample is assigned degree of membership values of each class based on membership values of its k nearest neighbors in those classes. Fuzzy K-NN resolves ties by using degree of memberships of neighbors to get degree of membership of sample in each of the classes available.

The Fuzzy k-nearest neighbor (k-NN) is a method was used for classification, a modified form of K- NN. It is a supervised / prototype based classification method. K- Nearest neighbors of unknown sample were selected initially and then class for that sample was selected by majority voting amongst them. The parameters in k-NN techniques are the number k which determines how many prototypes are to as the neighbors, and the distance function, generally the Euclidian distance is used. The problem with crisp K-NN is that how near or far a neighbor is does not matter until it is in k- nearest neighbors, it will have equal weight to other neighbors. Due to this even if a training sample lies just next to testing sample but all other majority nearest neighbors are at a longer distance compared, but the class of test sample will be determine on majority voting in which distant majority may win. Another problem with crisp K-NN is that if a tie situation occurs, the class is assigned arbitrarily to the lower class label. Moreover giving equal weights to all k- nearest neighbor

prototypes can introduces error, if there is noise in prototype (relative to the chosen value of K).

In Fuzzy K-NN an unknown sample is assigned membership to the class most represented by its K nearest neighbors, while giving a fuzzy weighting to the distance of neighbors. Fuzzy K-NN removes the crispness problem of crisp K-NN, and generally produces more reliable & accurate results. Degree of membership of sample x in i^{th} class is given by:

$$\mu_i(x) = \frac{\sum_{j=1}^{K} \mu_{ij} \left(\frac{1}{\|x - x_i\|^{\frac{2}{m-1}}} \right)}{\sum_{j=1}^{K} \left(\frac{1}{\|x - x_i\|^{\frac{2}{m-1}}} \right)} \qquad \text{Eq. 17}$$

There are two significant limits of values for m.

- m < 1: for smaller values of m, distant samples have greater influence in classification of an unknown sample. For -1<m<1, as m decreases, the influence increases exponentially.
- m > 1: for larger the values of m, distant samples have the lesser influence in classification. For 1<m<3 as m increases, the influence of distant samples decreases exponentially.
- As m approaches $\pm\infty$ the results of the classifications approach an estimation of crisp K-NN.

The values of k have an effect on noisy data, greater the value of k more robust the classification becomes. But that makes boundaries of classes fuzzy. The smaller values of k, makes distinct boundaries between classes but are less robust against noise. The cross validation was used to choose optimal values for k and m. Generally, heuristic techniques are used to choose optimal k and m, like cross validation or leave one out. Another popular approach is to use evolutionary algorithms to find optimal values for k and m. the Figure 3-14 shows the Nearest Neighbor with decision boundaries.

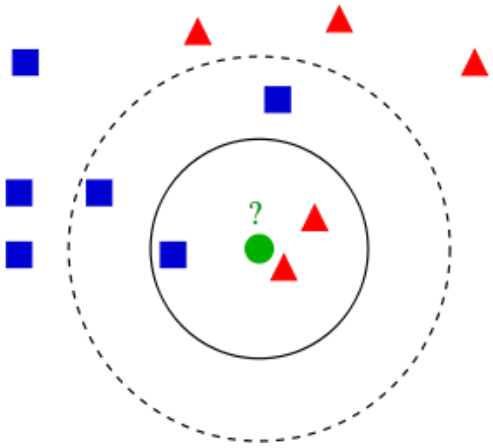


Figure 3-14 Nearest Neighbors with decision boundaries

Chapter 4. Results and Discussion

The results are calculated against the each given above techniques. These results are calculated using different parameters, which include size of the training samples, size of testing samples, threshold and fault tolerance.

4.1 Using KNN:

As we know that Discrete Cosine Transform is used to reduce the dimensionality of features vectors. The truncation of feature vector is done by passing a scalar number **n** to the input arguments of the **y = dct(x , n)**. n is a simple number that pads or truncates x to length n before transforming. This process of reducing the coefficients is necessary, as in case of classification we need a feature vector of fixed length for each signature. In our project DCT is used to reduce the feature set. Advantage of dct on other transform is that it provides a good compromise between information packing ability and computational complexity. The experiments are performed by using n = 7, 10, 13, 17… and a prominent difference can be seen in error.

Table 4-1 Error vs. Dimensionality Coefficient

Sr. No.	DCT dimensionality Coefficient	Error (%)
1.	7	69.70%
2.	10	65.40%
3.	13	59.80%
4	17	55.40%

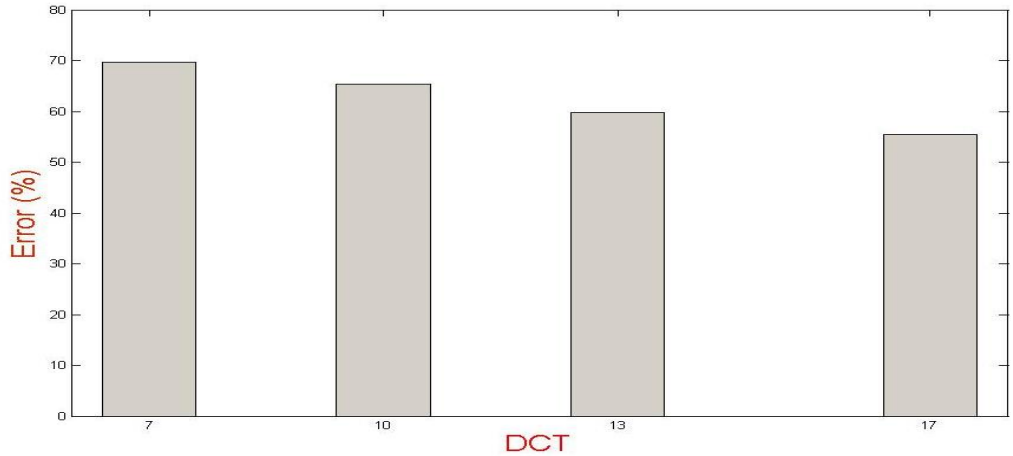


Figure 4-1 Histogram Error vs Dimensionality

Different combinations of local properties are used to reduce the error. The results of those experiments are shown in the Table 4-2and plot in Figure 4-2.

Table 4-2 Error analysis local property for DCT 10

Sr. No.	Local Properties used against DCT 10	Error (%)
1	x-coordinates, y-coordinates and pressure LP(123)	66.10%
2	x-coordinates, pressure and x-coordinate difference LP(134)	69.10%
3	x-coordinates, y-coordinates LP(12)	66.10%

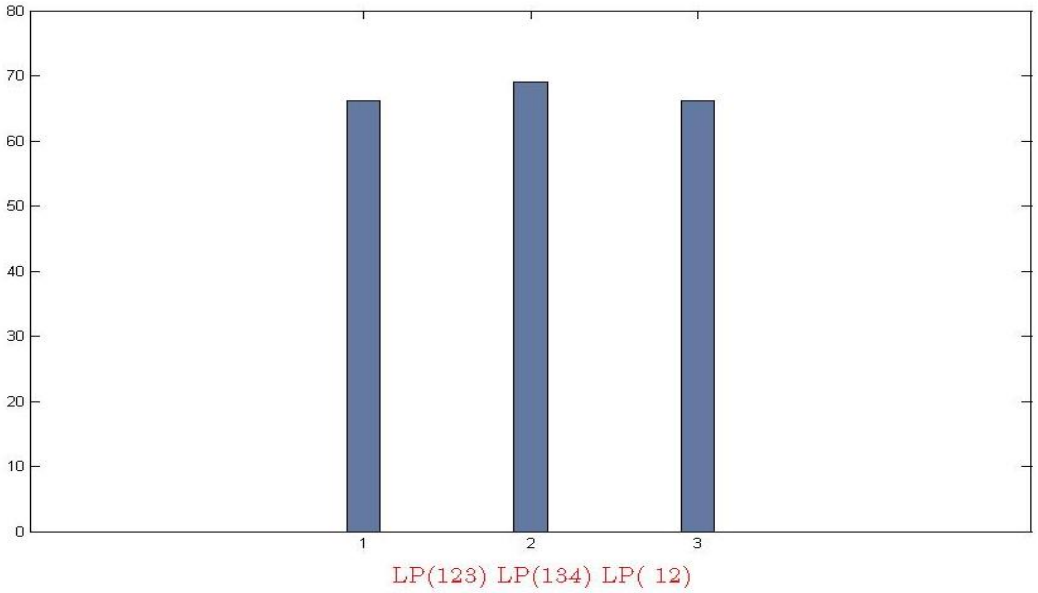


Figure 4-2 Error analysis local property for DCT 10

For DCT 17 using different local properties results obtained are shown in Table 4-3and plot in Figure 4-3.

Table 4-3 Error analysis of local properties for DCT 10

Sr. No.	Local Properties used against DCT 17	Error (%)
1	x-coordinates, y-coordinates , pressure and x-coordinate diff LP(1234)	50.60%
2	x-coordinates, y-coordinates and pressure LP(123)	50.60%
3	x-coordinates, dx and dy LP(145)	59.60%

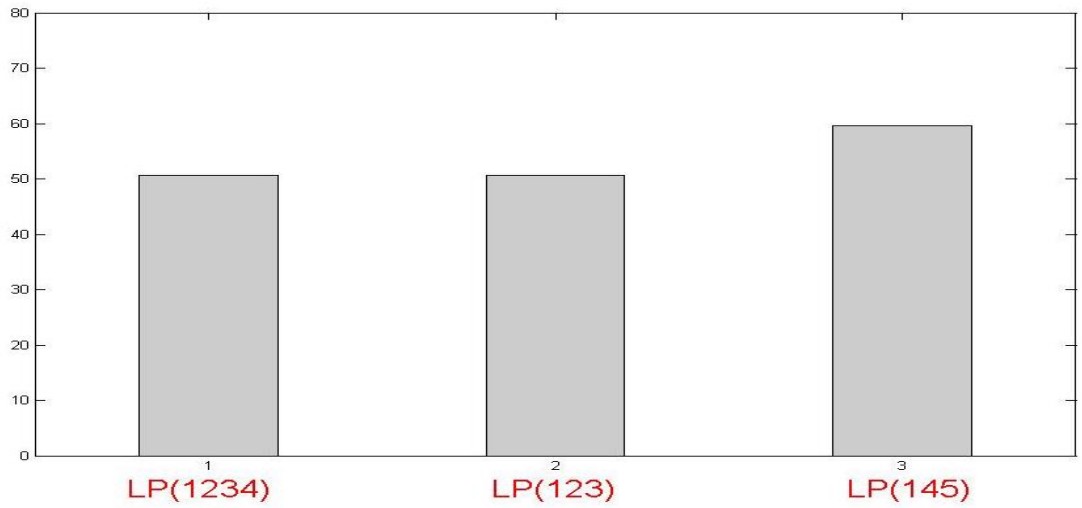


Figure 4-3 3Error Analysis using Different Local Properties for DCT 17

4.2 Using LPD:

In LPD we used different combinations of parameters. The training and testing samples are changed. The effects of normalization are observed on the results. As lpd has two parameters that are changed and the values are observed.

Table 4-4 Un-normalized Database Distribution for LPD

Database Distribution		
Training Samples	Testing Samples	Skilled Forgeries
10 per person	5 per person	5 per person

With un-normalized database the results obtained are shown in Table 4-5.

Table 4-5 Results with LPD un-normalized DB

NU	S	FRR(Random) %	FAR(Random) %	Accuracy(Skilled) %
0.4000	0.5000	0.1960	43.8000	94.4000
0.3500	1.0000	0.3111	39.0000	93.6000
0.3000	1.5000	0.3232	35.4000	93.2000
0.2500	0.5000	0.5455	32.4000	90.2000
0.2000	1.5000	1.0384	26.4000	87.4000
0.1000	3.0000	1.5232	23.8000	85.8000
0.0010	3.0000	16.8626	18.4000	74.8000

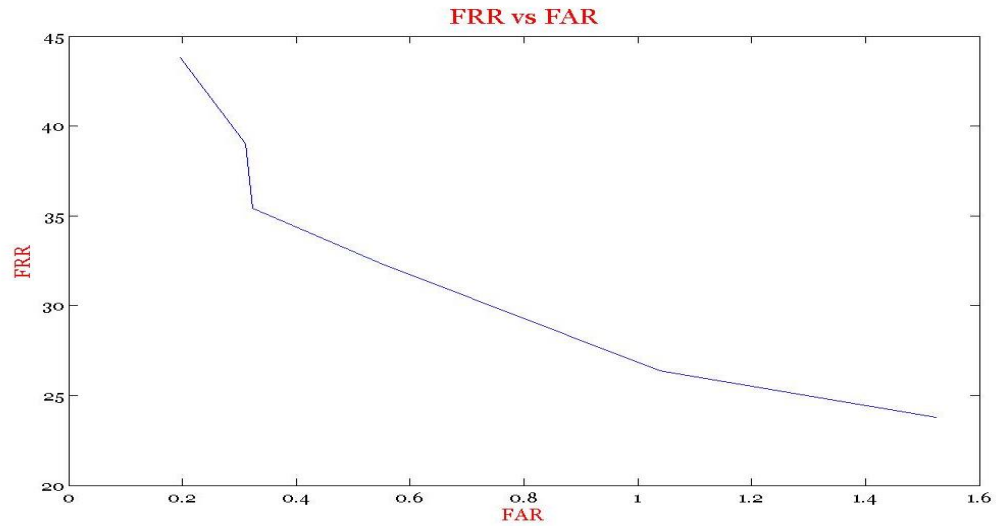


Figure 4-4 FAR vs FRR Un-normalized DB

Now with normalized database the results are calculated and shown in Table 4-6and Figure 4-5 shows its plot.

Table 4-6 Results with LPD normalized DB

NU	S	FRR(Random) %	FAR(Random) %	Accuracy(Skilled) %
0.40	0.50	0.49	42.40	92.60
0.30	0.50	0.91	33.40	90.60
0.15	1.5	3.34	24	83
0.01	2	3.34	24	83

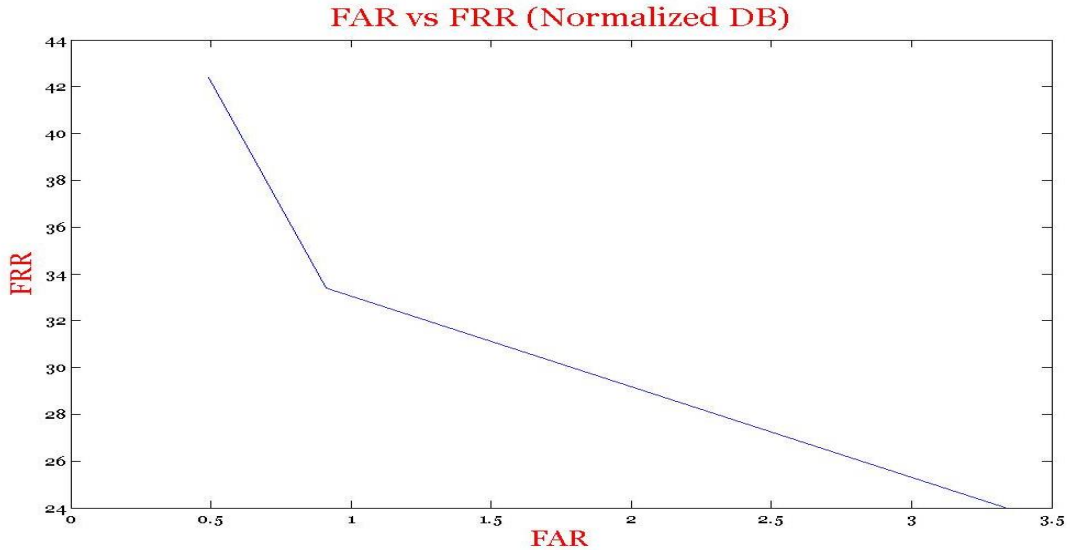


Figure 4-5 FAR vs FRR Normalized DB

4.3 Pruned Fuzzy k-Nearest Neighbor Classifier (Pfknn):

The results obtained using Pruned Fuzzy k-Nearest Neighbor (pfknn). The classification is done using different combination of the training and testing samples. The results obtained in this classification method were the best so for. The results are calculated using well known terms False Acceptance Ratio (FAR) and False Rejection Ratio (FRR). The FAR is the basically the number of False Acceptance to the total attempts, similarly the FRR is total number of False Rejections to the total Attempts.

In our classification we used the first combination of database as given below and there results are shown in Table 4-7.

Table 4-7 Results using PFKNN (trn = 600, tst = 900)

Training Samples = 600	Testing Samples = 900	Skilled Forgeries = 500

Threshold	**FAR %**	**FRR %**	**Accuracy(skilled) %**
12	4.66	3.22	58.4
11	4.55	5	69
10	3.77	7.88	77.4
9	3.44	13.11	84.6
8	2.11	23.6	89.6
7	1.11	38.44	93.8

The results shown in above table can be best described by using Figure 4-6.

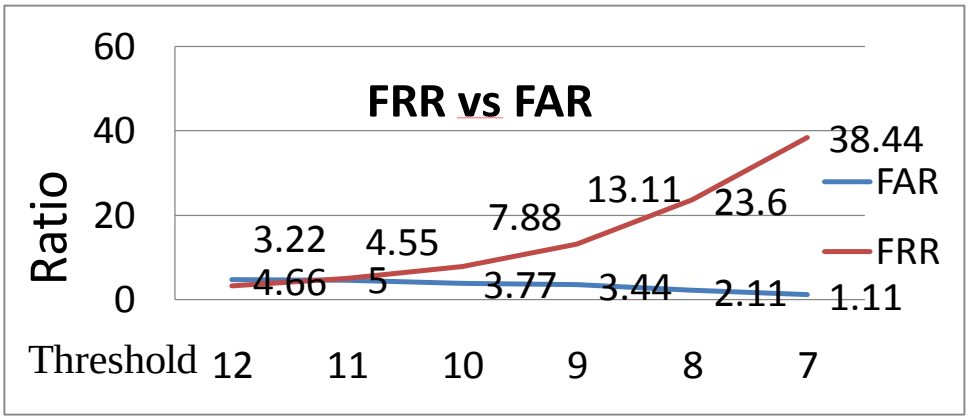


Figure 4-6 FRR vs. FAR using PFKNN (trn =600, tst = 900)

By using the 10 signature of each person for training and 5 for testing the results obtain in are shown in Table 4-8.

Table 4-8 Results using PFKNN (trn = 1000, tst = 500)

Training Samples = 1000	Testing Samples = 500	Skilled Forgeries = 500	
Threshold	**FAR %**	**FRR %**	**Accuracy(skilled) %**
12	2.8	1.4	55
11	2.8	2.0	65
10	2.0	5.0	74
9	1.8	7.4	82
8	1.4	14.2	87.2
7	1	26.4	91.8

The results in the above table can be best described using Figure 4-7.

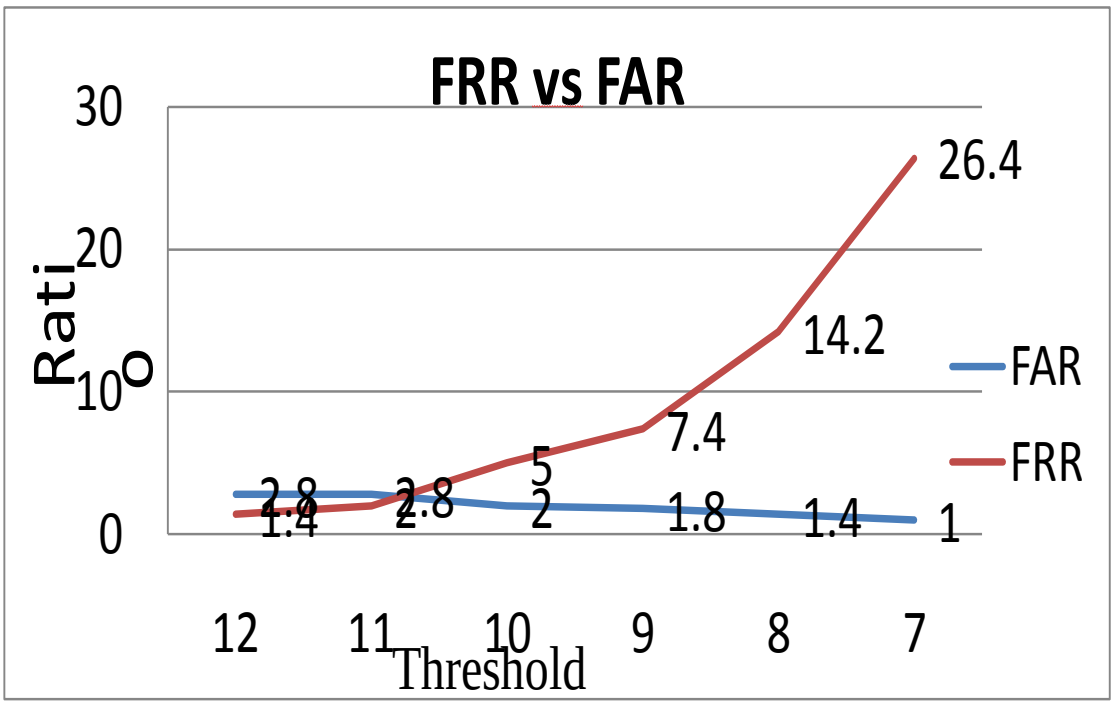


Figure 4-7 FRR vs. FAR using PFKNN (trn =1000, tst = 500)

By using the 9 signature of each person for training and 6 for testing the results obtained are shown in Table 4-9.

Table 4-9 Results using PFKNN (trn = 900, tst = 600)

Training Samples = 900	Testing Samples = 600	Skilled Forgeries = 500

Threshold	FAR %	FRR %	Accuracy(skilled) %
12	3.16	1.5	55
11	3.16	2.0	65
10	2.33	4.83	74.6
9	2.16	8.33	82
8	1.66	15.33	87.8
7	0.83	18.88	92.8

The results in the above table can be best described using this plot in Figure 4-8.

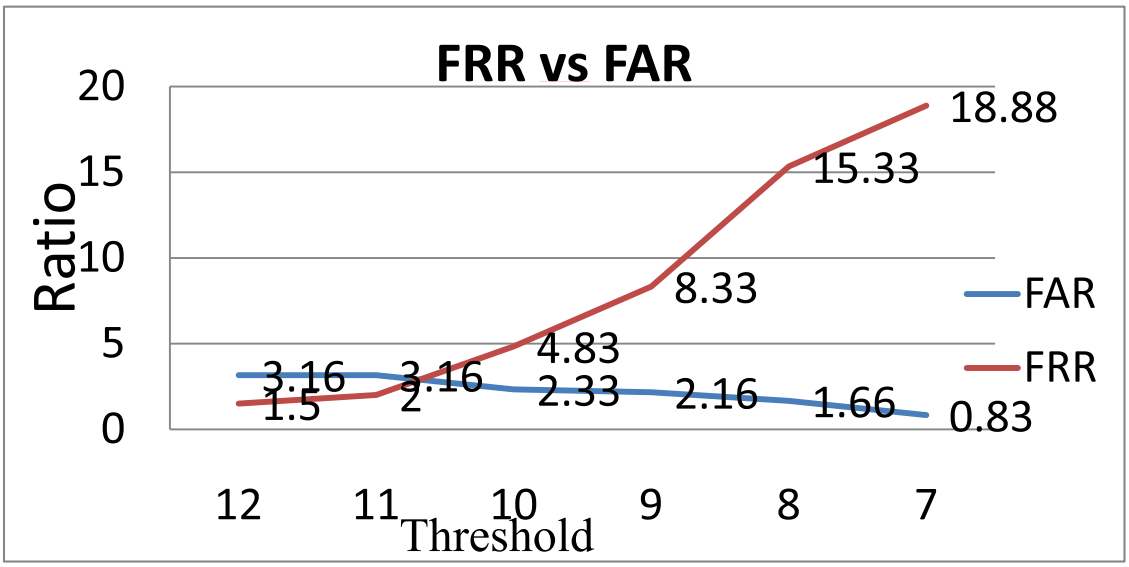


Figure 4-8 FRR vs. FAR using PFKNN (trn = 900, tst = 600)

4.4 Using Improved Knn:

The technique used for the classification of the signatures was modified. The classifier is then trained on 10 signatures per person. Now the template of each person will be generated using 10 signatures of that person. The testing of the signatures is done using the rest of 5 signatures; also the skilled forgeries are used to test the

system. The results are calculated using different threshold, which varied from 7-12. The distribution of the database sued for knn is shown in table given below:

Table 4-10 Database Distribution for knn (k = 4)

Database Distribution		
Training Samples	Testing Samples	Skilled Forgeries
10 per person	5 per person	5 per person

The results using the improved knn are given below.

Table 4-11 Results Using Improved Knn (k = 4)

Threshold	FRR %	FAR %	Accuracy (Skilled Forgeries) %
7	26.2000	2.8000	92.6000
8	14.4000	3.2000	87.2000
9	7.4000	3.6000	81.6000
10	4.8000	3.8000	75.6000
11	1.6000	4.6000	66.8000
12	1.2000	4.6000	56.8000

The results concluded that the good ratio of FAR and FRR can achieved if we have the threshold of 7 and 8. The system is demonstrated using threshold of 7 and 8. The figure below shows the results calculated using Improved knn.

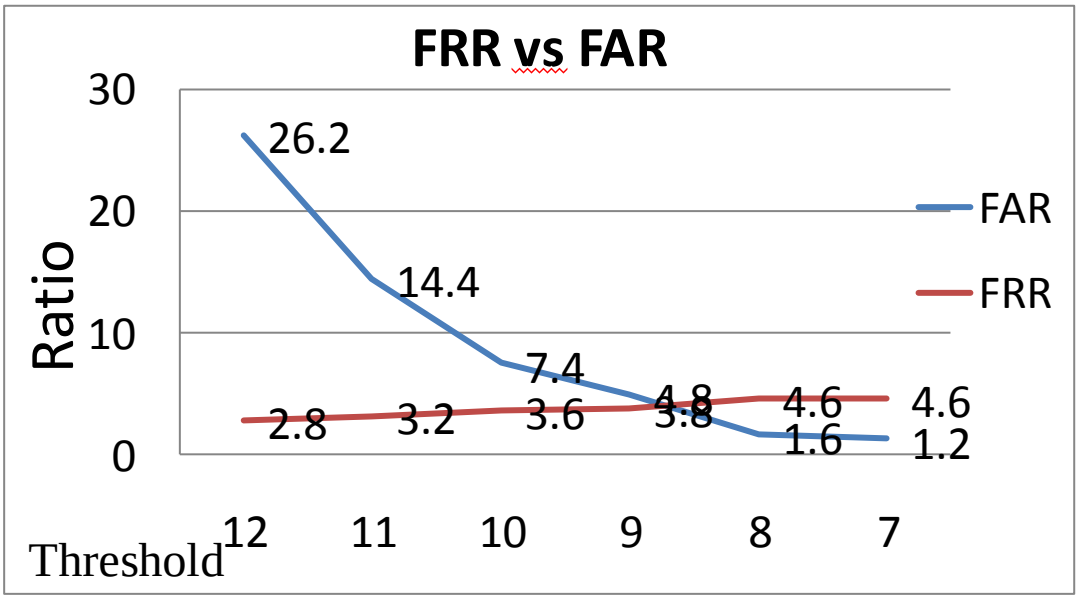


Figure 4-9 Results Using Improved Knn (k =4)

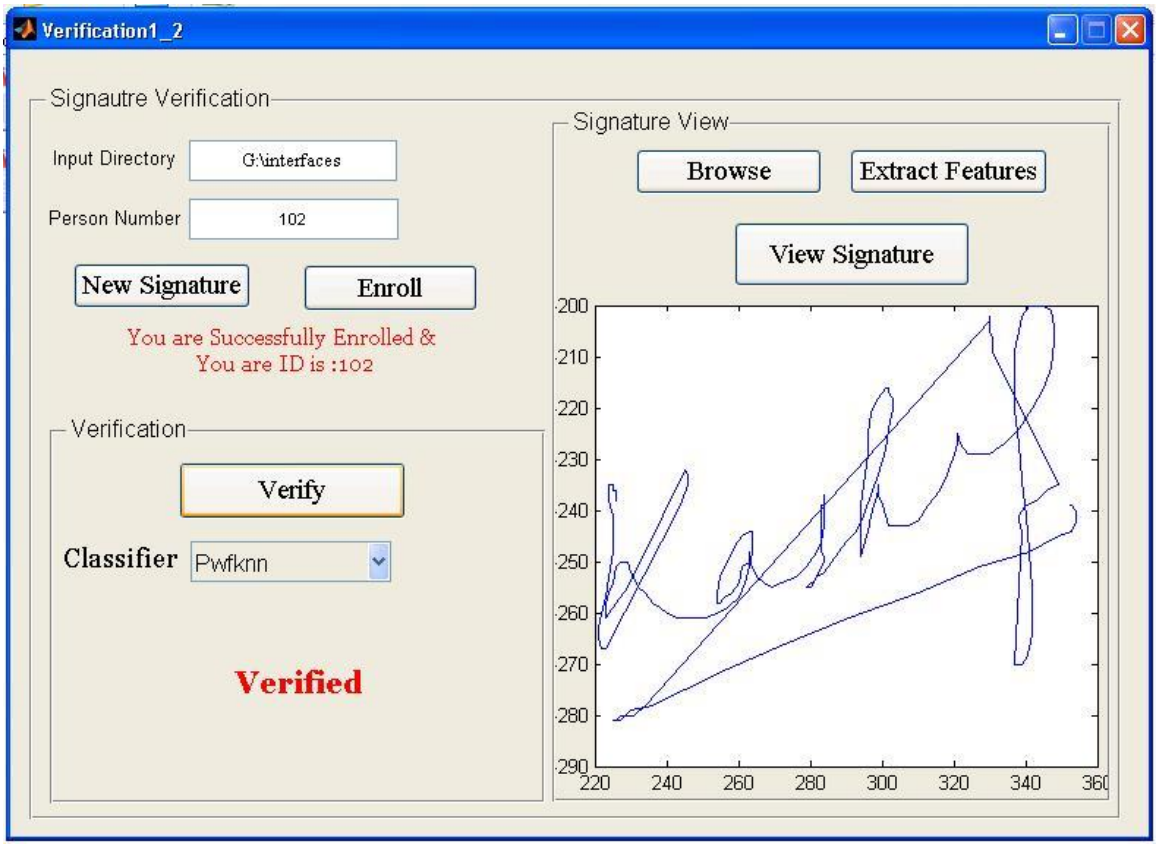
Chapter 5. Interfaces

The purpose of the project is to develop a system which verify signer on the basis of personal signatures. In order to fulfill these requirements two interfaces were developed. The first interface takes person id and the signatures and performs verification procedure. The interface also includes the view signature options, so that one can view signatures. The snapshot taken of the interface is shown below:

The working procedure of this interface is that the first enter the person id, then using browse button select the signature that needs to be verified against the desired person id. Extract feature button extract the features of that signature on spot. View signature button is used for signature visibility. The classification is done by two classifiers simple knn and pwfknn, one can select any of classifier and perform classification.

The second interface is a bit different in a sense that it is used for on spot enrollment of the signatures. It basically attaches an exe file with that interface so that

one can enroll himself /herself in database. The snapshot of the interface is given below:



The second interface is made so dynamically that one can take database, codes and interfaces anywhere and can deploy that system on any ordinary computer, just he has to input directory and the system will work accordingly.

Chapter 6. Conclusions and Future Recommendations

In this portion of thesis we will summarize the work done in this project. The research work is done under the heading of online signature verification; the purpose of this project was to develop the system for online signature verification.

6.1 Achievements:

At the end of this thesis we can say that:

- Research work has been done in field of online signatures verification.
- Study and Implementation of Wavelet and Cosine Transform
- Classification techniques including k-Nearest Neighbor, Linear programming Descriptor(lpd) and Pruned Fuzzy k Nearest Neighbor (PFKNN).
- Optimization of these techniques to get better results

6.2 Conclusions:

From the above written thesis and work performed we can conclude the following things.

Signature verification is an important biometrics technique commonly used. From our experiments we found that the accuracy mainly depends on the features selected, the transform applied for extraction and reduction of these features and the main thing how one classify them. As from experiments we can say the classification method is the most important thing in signature verification. In our case we used three methods for classification of signatures and we have seen that by changing the approach and classifier we got better results.

In case of knn the results was worst. For that case we got error more than 50% in that case we have wrong features used for classification and the approach was also wrong in that case. But this classifier is used by most of the researchers and got best results using that classifier. using knn the error occurred was 45%. By using improved knn the results I got is much better than the other classifiers. The FAR for random

forgeries 14.40% and the FRR for Random forgeries is 3.2 %, the skilled forgeries resulted the FAR of 12.80%.

In using linear programming descriptor (lpd) we found that it's much better technique and resulted in much better accuracy and results. The classifier results in the FAR of 14.20% on skilled forgeries, with FRR 1.52% and FAR of 23.80%.

In using Pruned Fuzzy k Nearest Neighbor (PFKNN) the results were the best. The classifier was trained and tested using different combination. The results calculated on the each combination and best results found are the FAR of 15.4%, with FAR 3.44% and FRR 13.11%.

We have seen that the combination of the six local properties of Nanni [9] used with other nine local properties mentioned in Afsar [10] resulted in the best results. So the combinations of local properties are defined as best of them.

The results of knn classifier showed that the local properties like pressure difference and x-coordinate difference resulted in greater error.

6.3 Recommendations and Future Work

The recommendation and future work in this field are described in following way:

- **Reduction of Database Errors:** The error in database can result in most of faults. So if we have an accurate database then we have much better results. The main problem in signature verification is that the signatures of even a single person varies with time and we don't have an accurate method to gather error free database.

- **Reduction of signature space:** If we reduce signature space then this may result in small noise in signatures. This will also reduce feature extraction time and also reduce error in classification.

- **Use of best possible combinations:** in using the combination of local properties one have to select the best possible combination because in most of cases we have some local properties

- **Visual Feedback and sampling rate:** The tablet or the digital device used for capturing signatures should have a high sampling rate to capture data more accurately. In case of tablet used for performing and capturing signatures, there should be visual feedback in the tablet so as the signer can see while he/she is signing. This helps a lot because it usually confuses the signer when he/she signs on a surface where nothing is drawn. This will also save a lot of time taken by signers to adapt to a tablet with no visual feedback. A device recording pen tilt will also help classifying a signature.

References

[1]. Wikipedia, ”Daubechies wavelet” [Online] Available: http://en.wikipedia.org/wiki/Daubechies_wavelet

[2]. Wikipedia, ”Discrete wavelet transform”[Online] Available: http://en.wikipedia.org/wiki/Discrete_wavelet_transform

[3]. Scholarpedia, “K-nearest_neighbor” [Online] Available: http://www.scholarpedia.org/article/K-nearest_neighbor

[4]. Wikipedia, “K-nearest_neighbor_algorithm” [Online] Available: http://en.wikipedia.org/wiki /K-nearest_neighbor_algorithm

[5]. Charles E. Pippin, “Dynamic Signature Verification using Local and Global Features”, Georgia Institute of Technology, July 2004.

[6]. Mohammad M. Shafiei and Hamid R. Rabiee, “An On-Line Signature Verification Algorithm Using Variable Length Segmentation and Hidden Markov Models”, ICDAR vol. 1, pp. 443-446, 2003.

[7]. R. S. Kashi , J. Hu & W. L. Nelson, “On-line Handwritten Signature Verification using Hidden Markov Model Features” , ICDAR ,pp. 253 – 257, 1997

[8]. Hao Feng and Chan Choong Wah, “Online signature verification using new extreme points warping technique”, PRL (24), No. 16, pp. 2943-2951, December 2003.

[9]. Loris Nanni, Alessandra Lumini, “A novel local on-line signature verification system”, DEIS, IEIIT – CNR, Universita` di Bologna, Viale Risorgimento 2, 40136 Bologna, Italy, 12 October 2007.

[10]. F.A. Afsar, M. Arif and U. Farrukh, “Wavelet Transform Based Global Features for Online Signature Recognition”, Department of Computer & Information Sciences Pakistan Institute of Engineering & Applied Sciences P.O. Nilore, Islamabad 45650, Pakistan July, 2006.

[11]. Alisher Anatolyevich Kholmatov, “Biometric Identity Verification Using On-Line & Off-Line Signature Verification”, Submitted to the Graduate School of Engineering and Natural Sciences in partial fulfillment of the requirements for the degree of Master of Science Sabanci University, Spring 2003.

[12]. E. Pekalska, D.M.J. Tax, and R.P.W. Duin. “One-class LP classifier for dissimilarity representations”, S. Becker, S. Thrun, and K. Obermayer, editors, Advances in Neural Information Processing Systems, volume 15. MIT Press: Cambridge, MA, 2003.

[13]. D.M.J. Tax, “PRTools 4.1 Manual”, D.M.J.Tax@prtools.org Faculty EWI, Delft University of Technology P.O. Box 5031, 2600 GA Delft, The Netherlands.

[14]. D.M.J. Tax “dd_tools Manual”, Faculty EWI, Delft University of Technology, The Netherlands, September 24, 2008.

[15]. Liang Wan, Bin Wan, Zhou-Chen Lin “On-Line Signature Verification With Two-Stage Statistical Models”, The Chinese University of Hong Kong lwan@cse.cuhk.edu.hk, Huazhong University of Science and Technique wanbin@gmail.com, Microsoft Research, Asia, zclin@microsoft.com .

[16]. Alisher Kholmatov and Berrin Yanikoglu “Biometric Authentication using Online Signatures”, alisher@su.sabanciuniv.edu, berrin@sabanciuniv.edu, http://fens.sabanciuniv.edu Sabanci University, Tuzla, Istanbul, Turkey 34956

[17]. Cman F. Lam, David Kamins and Kuno Zimerann, “Signature Recognition through Spectral Analysis”, DEFT. OF BIOMETRY MEDICAL UNIV. OF SC, DEPT. OF ELECT. ENGINEERING UNIV. OF MISSOURI.

[18]. F.A. Afsar, M. Arif, M. U. Akram and J. Khurshid “A Pruned Fuzzy k-Nearest Neighbor Classifier with Application to Electrocardiogram Based Cardiac Arrhytmia Recognition”, Department of Computer & Information Sciences Pakistan Institute of Engineering & Applied Sciences P.O. Nilore, Islamabad 45650, Pakistan.